

# **TINA**

## **Design Suite v7**

**The Complete Electronics Lab for Windows**

**QUICK START MANUAL**

**DesignSoft**

[www.designsoftware.com](http://www.designsoftware.com)

## COPYRIGHTS

© Copyright 1990-2006 DesignSoft, Inc. All rights reserved.

All programs recorded on the original release CD of *TINA* and the accompanying documentation are copyrighted. *TINA* is provided under a License Agreement and may be used or copied only in accordance with its terms and conditions.

## LIMITED LIABILITY

*TINA*, together with all accompanying materials, is provided on an “as is” basis, without warranty of any kind.

DesignSoft, Inc., its distributors, and dealers make no warranty, either expressed, implied, or statutory, including but not limited to any implied warranties of merchantability or fitness for any purpose.

In no event will DesignSoft Inc., its distributor or dealer be liable to anyone for direct, indirect, incidental or consequential damages or losses arising from the purchase of *TINA* or from use or inability to use *TINA*.

## TRADEMARKS

*IBM PC/AT, PS/2* are registered trademarks of *International Business Machines Corporation*

*Windows, Windows 9x/ME/NT/2000/XP* are trademarks of *Microsoft Corporation*

*PSPice* is a registered trademark of *MicroSim Corporation*

*Corel Draw* is a registered trademark of *Corel Inc.*

*TINA* is a registered trademark of *DesignSoft, Inc.*

# TABLE OF CONTENTS

<b>1.</b>	<b>INTRODUCTION</b>	<b>1-1</b>
1.1	What is TINA and TINA Design Suite? .....	1-1
1.2	Available Program Versions .....	1-6
1.3	Optional supplementary hardware .....	1-7
1.3.1	TINALab II High Speed Multifunction PC Instrument .....	1-7
<b>2.</b>	<b>NEW FEATURES IN TINA</b>	<b>2-1</b>
2.1	List of new features in TINA 7.0 .....	2-1
2.2	List of new features in TINA PRO 6 .....	2-3
2.3	List of new features in TINA PRO 5.5 .....	2-4
<b>3.</b>	<b>INSTALLATION AND START-UP</b>	<b>3-1</b>
3.1	Installation Procedure .....	3-1
3.1.1	Minimum hardware and software requirements .....	3-1

- 3.1.2    Installation from CD-ROM ..... 3-2
- 3.1.3    Following the Installation Steps ..... 3-3
- 3.1.4    Welcome and Software License Agreement ..... 3-3
- 3.1.5    Entering User Information ..... 3-3
- 3.1.6    Choose Destination Location ..... 3-4
- 3.1.7    Selecting a Setup Type ..... 3-5
  - 3.1.7.1    Typical ..... 3-5
  - 3.1.7.2    Compact ..... 3-6
  - 3.1.7.3    Custom ..... 3-6
- 3.1.8    Selecting the Program Folder ..... 3-6
- 3.1.9    Selecting the Symbol Set ..... 3-6
- 3.1.10   Final check and copying the files ..... 3-7
- 3.1.11   Completing the Setup ..... 3-7
- 3.2    Uninstalling TINA ..... 3-8
- 3.3    Network Installation ..... 3-8
- 3.4    Copy Protection ..... 3-10
  - 3.4.1    Copy Protection by Software ..... 3-10
  - 3.4.2    Copy Protection by Hardware (dongle) ..... 3-10
- 3.5    Starting Up ..... 3-10
- 3.6    Experimenting with Example Circuits, avoiding common problems ..... 3-11

**4.    GETTING STARTED    4-1**

---

- 4.1    Schematic Editing Using the Mouse ..... 4-1
  - 4.1.1    Using the right mouse button ..... 4-1
  - 4.1.2    Using the left mouse button ..... 4-2

4.2	Measurement Units .....	4-4
4.3	The Basic Screen Format .....	4-4
4.4	Placing the Circuit Components .....	4-11
4.4.1	Wire .....	4-12
4.4.2	Input and Output .....	4-13
4.5	Exercises .....	4-14
4.5.1	Editing an RLC Circuit Schematic .....	4-14
4.6	Analyses .....	4-18
4.6.1	Analyzing an RLC Circuit (DC, AC Transient and Fourier analysis .....	4-18
4.6.2	Creating and analyzing an OP-AMP circuit .....	4-27
4.6.2.1	Calculating DC Transfer characteristic .....	4-32
4.6.3	Analysis of SMPS circuits .....	4-33
4.6.4	Stress Analysis .....	4-43
4.6.5	Network Analysis .....	4-44
4.6.6	Analyzing a Digital Circuit with TINA's Digital Engine .....	4-45
4.6.7	Analyzing a Digital Circuit using Digital VHDL Simulation .....	4-49
4.6.8	Testing Your Circuit with Virtual and Real Time Instruments .....	4-53
4.6.9	Testing your circuit in interactive mode .....	4-55
4.6.9.1	Digital Circuit with a Keypad .....	4-56
4.6.9.2	Light Switch with Thyristor .....	4-57
4.6.9.3	Ladder Logic networks .....	4-58
4.6.9.4	VHDL Circuits .....	4-59
4.6.9.5	Microcontroller ( MCU ) Circuits .....	4-61
4.6.9.6	Example PIC Flasher .....	4-63
4.6.9.7	Example PIC Interrupt handling .....	4-65
4.6.9.8	Editing the Code in the Debugger .....	4-69
4.6.9.9	Making a Breakpoint .....	4-70

## 5. USING SUBCIRCUITS, SPICE MACROS AND S-PARAMETERS

## 6. MAKING YOUR OWN SCHEMATIC SYMBOLS AND FOOTPRINTS 6-1

i-6 TINA Quick Start

<b>7.</b>	<b>USING THE PARAMETER EXTRACTOR</b>	<b>7-1</b>
-----------	--------------------------------------	------------

---

<b>8.</b>	<b>ADVANCED TOPICS</b>	<b>8-1</b>
-----------	------------------------	------------

---

8.1	Introduction .....	8-1
-----	--------------------	-----

8.2	Table of Contents .....	8-2
-----	-------------------------	-----





## CHAPTER 1

# INTRODUCTION

## 1.1 What is TINA and TINA Design Suite?

---

TINA Design Suite is a powerful yet affordable software package for analyzing, designing and real time testing of analog, digital, VHDL, and mixed electronic circuits and their layouts. You can also analyze RF, communication, and optoelectronic circuits, and test and debug microprocessor & microcontroller applications. A unique feature of TINA permits you to bring your circuit to life with the optional USB controlled TINALab II hardware. TINALab II turns your computer into a powerful, multifunction T&M instrument. Electrical engineers will find TINA an easy to use, high performance tool, while educators will welcome its unique features for the training environment.

Starting with v7, TINA is distributed in two major versions— TINA v7 and TINA Design Suite v7 . **TINA v7** includes circuit simulation only, while **TINA Design Suite v7** also includes the advanced PCB designer. This fully integrated layout module has all the features you need for advanced PCB design, including Multilayer PCB's with split power planes, powerful autoplacement & autorouting, rip-up and reroute, manual and "follow-me" trace placement, DRC, forward and back annotation, pin and gate swapping, keep-in and keep-out areas, thermal relief, fanout, plane layers, Gerber file output and much more.

TINA can also be used in the training environment. It includes unique tools for testing students' knowledge, monitoring progress and introducing troubleshooting techniques. With optional hardware it can be used to test real circuits for comparison with the results obtained from simulation. Of great importance to educators, the package includes all the tools needed to prepare educational materials.

**Schematic Capture.** Circuit diagrams are entered using an easy to use schematic editor. Component symbols chosen from the Component bar are positioned, moved, rotated and/or mirrored on the screen by the mouse. TINA's semiconductor catalog allows the user to select components from a user-extendible library. An advanced “rubber wire” tool is provided allowing easy modification of the schematic diagrams. You can open any number of circuit files or subcircuits, cut, copy and paste circuit segments from one circuit into another, and, of course analyze any of the currently open circuits. TINA gives you tools to enhance your schematic by adding graphics elements such as lines, arcs, arrows, frames around the schematic, and title blocks. You can also draw non-orthogonal (diagonal) components such as bridges and 3-phase networks.

**PCB Design.** TINA v7 includes only circuit simulation, while TINA Design Suite v7 includes TINA's advanced PCB designer. This fully integrated layout module has all the features you need for advanced PCB design, including Multilayer PCB's with split power planes, powerful autoplacement & autorouting, rip-up and reroute, manual and “follow-me” trace placement, DRC, forward and back annotation, pin and gate swapping, keep-in and keep-out areas, thermal relief, fanout, plane layers, 3D view from any angle, Gerber file output and much more.

**Electrical Rules Check (ERC)** will examine the circuit for questionable connections between components and display the results in the Electrical Rules Check window. ERC is invoked automatically, so missing connections will be brought to your attention before analysis begins.

**Schematic Symbol Editor.** In *TINA*, you can simplify a schematic by turning portions of it into a subcircuit. In addition, you can create new *TINA* components from any Spice subcircuit, whether created by yourself, downloaded from the Internet, or obtained from a manufacturer's CD. *TINA* automatically represents these subcircuits as a rectangular block on your schematic, but you can create any shape you like with TINA's Schematic Symbol Editor.

**Library Manager.** *TINA* has large libraries containing Spice- and S-parameter models provided by semiconductor manufacturers such as Analog Devices, Texas Instruments, National Semiconductor, and others. You can add more models to these libraries or create your own Spice- and S-parameter library using TINA's Library Manager (LM).

**Parameter Extractor.** Using TINA's Parameter Extractor you can also create component models that more closely represent actual real world devices by converting measurement or catalog data into model parameters.

**Text and Equation Editor.** *TINA* includes a Text and Equation Editor for annotating schematics, calculations, includes graphic output, and measurement results. It is an invaluable aid to teachers preparing problems and examples.

The circuit diagrams and the calculated or measured results can be printed or saved to files in standard Windows BMP, JPG and WMF format. These output files can be processed by a number of well known software packages (Microsoft Word, Corel Draw etc.). Netlists can be exported and imported in Pspice format and also to drive popular PCB packages such as ORCAD, TANGO, PCAD, PROTEL, REDAC and other programs.

**DC analysis** calculates the DC operating point and the transfer characteristic of analog circuits. The user can display the calculated and/or measured nodal voltages at any node by selecting the node with the cursor. For digital circuits, the program solves the logic state equation and displays the results at each node step-by-step.

**Transient analysis.** In the transient and mixed mode of TINA you can calculate the circuit response to the input waveforms that can be selected from several options (pulse, unit step, sinusoidal, triangular wave, square wave, general trapezoidal waveform, and user-defined excitation) and parameterized as required. For digital circuits, programmable clocks and digital signal generators are available.

**Fourier analysis.** In addition to the calculation and display of the response, the coefficients of the **Fourier series**, the **harmonic distortion** for periodic signals, and the **Fourier spectrum** of non-periodic signals can also be calculated.

**Digital Simulation.** TINA also includes a very fast and powerful simulator for digital circuits. You can trace circuit operation step-by-step, forward and backward, or view the complete time diagram in a special logic analyzer window. In addition to logic gates, there are ICs and other digital parts from TINA's large component library.

**VHDL simulation.** TINA now includes an integrated VHDL simulator to verify VHDL designs both in digital and mixed-signal analog-digital environments. It supports the IEEE 1076-1987 and

1076-1993 language standards and IEEE standards 1164 (standard logic). Your circuits can contain editable VHDL blocks from TINA's library, FPGAs & CPLDs, or VHDL components created by yourself or downloaded from the Internet. You can edit the VHDL source of any VHDL component and see the result instantly. TINA includes a wide range of PIC microcontroller models which you can test, debug and run interactively. The built in MCU assembler allows you to modify your assembler code and see the result promptly. Other MCU models including 8051 and AVR; more are coming soon. With the optional external VHDL simulator, you can develop and debug your VHDL code both externally and inside TINA. The VHDL simulator includes Waveform Display, Project Management and Hierarchy Browser, and 64-bit time.

**AC analysis** calculates, complex voltage, current, impedance, and power can be calculated. In addition, Nyquist and Bode diagrams of the amplitude, phase and group delay characteristics of analog circuits can be plotted. You can also draw the complex **phasor diagram**. For non-linear networks, the operating point linearization is done automatically.

**Network analysis** determines the two-port parameters of networks (S, Z, Y, H). This is especially useful if you work with **RF circuits**. Results can be displayed in Smith, Polar, or other diagrams. The network analysis is carried out with the help of TINA's network analyzer. The RF models of the circuit elements can be defined as SPICE subcircuits (SPICE macros) which contain parasitic components (inductors, capacitors) or as an S-parameter model defined by its S (frequency) function. S functions are normally provided by the component manufacturers (based on their measurements) and can be downloaded from the Internet and inserted into TINA either manually or by using TINA's library manager.

**Noise analysis** determines the noise spectrum with respect to either the input or the output. The noise power and the signal-to-noise ratio (SNR) can also be calculated.

**Symbolic analysis** produces the transfer function and the closed form expression of the response of analog linear networks in DC, AC, and transient modes. The exact solution, calculated through the symbolic analysis, can also be plotted and compared to the numerically calculated or measured results. The built-in **interpreter** can evaluate and plot arbitrary functions.

**Monte-Carlo and Worst-case analysis.** Tolerances can be assigned to the circuit elements for use in Monte-Carlo and/or worst-case analyses. The results can be obtained statistically, and their expected means, standard deviations and yields can also be calculated.

**Optimization.** TINA's enhanced optimization tool can tweak one or more unknown circuit parameters to achieve a predefined target response. The target circuit response (voltage, current, impedance, or power) must be “monitored” by meters. For example, you can specify several working point DC voltages or AC transfer function parameters and have TINA determine the values of the selected components.

**Post-processor.** Another great new tool of TINA is its post-processor. With the post-processor, you can add new curves of virtually any node and component voltage or current to existing diagrams. In addition, you can post-process existing curves by adding or subtracting curves, or by applying mathematical functions to them. You can also draw **trajectories**; i.e., draw any voltage or current as a function of another voltage or current.

**Presentation.** With TINA you can make quality documents incorporating **Bode plots, Nyquist, Phasor, Polar and Smith diagrams, transient responses, digital waveforms** and other data using linear or logarithmic scales. Customize presentations easily using TINA's advanced drawing tools—you can print your plots directly from TINA, cut and paste them into your favorite word processing package, or export them to popular standard formats. Customization includes complete control over texts, axes, and plot style; e.g., setting line width and color, fonts in all sizes and color, and automatic or manual scaling for each axis.

**Interactive mode.** When everything is in order, the ultimate test of your circuit is to try it in a “real life” situation using its interactive controls (such as keypads and switches) and watching its displays or other indicators. You can carry out such a test using TINA's interactive mode. You can not only play with the controls, but you can also change component values while the analysis is in progress. In addition, you can assign hotkeys to component values and switches to change them simply by pressing a key. You will immediately see the effect of the change. You can also test MCU applications in TINA's interactive mode. You can not only run and test them using the several lifelike interactive controls e.g., keyboards, but you can also debug them while the MCU executes ASM code step by step. And displays the

register contents and TINA's outputs in each step. If necessary you can modify the ASM code on the fly and test your circuit again without using any other tool.

**Virtual instruments.** In addition to standard analysis presentations such as Bode and Nyquist plots, *TINA* can present its simulation results on a wide range of high-tech virtual instruments. For example, you can simulate the time response of your circuit using a virtual square wave generator and a virtual oscilloscope. Using TINA's virtual instruments is a good way to prepare for the use of real test and measurement equipment. Of course it is important to remember that the "measurement results" obtained with virtual instruments are still simulated.

**Real-time Test & Measurements.** *TINA* can go beyond simulation when **supplementary hardware** is installed on the host computer. With this hardware, TINA's powerful tools can make **real-time measurements** on real circuits and display the results on its virtual instruments.

**Training and Examination.** *TINA* has special operating modes for training and for examination. In these modes, under TINA's control, the students solve problems assigned by the teacher. The solution format depends on the types of problems: they can be selected from a list, calculated numerically, or given in symbolic form. The interpreter - providing a number of solution tools - can also be used for problem solving. If the student cannot solve the problem, he/she can turn to the multilevel Advisor. The package includes all the tools needed to produce educational materials. A collection of examples and problems worked out by teachers is also part of the package. Another special educational function of *TINA* is the **software or hardware simulation of circuit faults** to practice troubleshooting. Using *TINA*, you can transform existing PC classrooms into contemporary electronics training labs at low cost.

## 1.2 Available Program Versions

---

Different program versions, tailored to meet various needs, are available.

Starting with v7, TINA is distributed in to major versions TINA v7 and TINA Design Suite v7. TINA v7 includes simulation only while TINA Design Suite v7 includes our new advanced PCB designer too.

Both versions are available with the following features:

- **Industrial version:** Includes all of TINA's features and utilities.
- **Network version:** The program can be used under Novell Netware version 3.12 or later, as well as under any Microsoft Network (Win9x/ ME/NT/2000/XP). It is especially recommended for corporate and classroom use.
- **Educational version:** It has most features of the Industrial version but parameter stepping and optimizations are allowed for one parameter only, Stress Analysis and the Steady State Solver are not included.
- **Classic Edition:** It has the same features as the Educational version above, except that Network Analysis is not allowed, TINA's large S-parameter component library and the Parameter Extractor, the External VHDL simulator utility, Stress Analysis and the Steady State Solver are not included.
- **Student Version:** Has the same features as Classic Edition version except that the circuit size is limited to 100 nodes including internal Spice macro nodes. The number of pads on the PCB layout is also limited to 100.
- **Basic version:** Has the same features as Classic Edition except that the circuit size is limited to 100 nodes including internal Spice macro nodes. The number of pads on the PCB layout is also limited to 100.

## 1.3 Optional supplementary hardware

### 1.3.1 TINALab II High Speed Multifunction PC Instrument

With TINALab II you can turn your laptop or desktop computer into a powerful, multifunction test and measurement instrument. Whichever instrument you need; multimeter, oscilloscope, spectrum analyzer, logic analyzer, arbitrary waveform generator, or digital signal

generator it is at your fingertips with a click of the mouse. In addition TINALab II can be used with the TINA circuit simulation program for comparison of simulation and measurements as a unique tool for circuit development, troubleshooting, and the study of analog and digital electronics.

TINALab II includes a DC to 50MHz bandwidth, 10/12 bit resolution, dual-channel **Digital Storage Oscilloscope**. Due to its advanced equivalent-time sampling technology, TINALab can acquire any repetitive signal with up to **4GS/s equivalent sampling rate**, while in single shot mode the sampling rate is 20 MS/s. The full scale input range is  $\pm 400V$ , with 5mV to 100V/div ranges.

The synthesized **Function Generator** provides sine, square, ramp, triangle and arbitrary waveforms from DC to 4MHz, with logarithmic and linear sweep, and modulation up to 10V peak to peak. Arbitrary waveforms can be programmed via the high level, easy to use language of TINA's Interpreter. Working automatically in conjunction with the Function Generator, the **Signal Analyzer** measures and displays Bode amplitude and phase diagrams, Nyquist diagrams, and also works as **Spectrum Analyzer**.

Digital I/O for the high-tech **Digital Signal Generator** and Logic Analyzer instruments allow fast 16-channel digital testing up to 40MHz.

The optional **Multimeter** for TINALab II allows DC/AC measurements in ranges from 1mV to 400V and 100 mA to 2A. It can also measure DC resistance in ranges from  $1\Omega$  to  $10M\Omega$ .

You can also plug **Experimenter Modules** into the slot on the front of TINALab II, allowing you to simulate, measure, and troubleshoot virtually the whole range of analog and digital electronics.

Using TINALab II with TINA gives you the unique capability to have circuit simulation and real time measurements in the same integrated environment. This provides an invaluable tool for troubleshooting and brings your designs to life by comparing simulated and measured results.



## CHAPTER 2

# NEW FEATURES IN TINA

This chapter describes the new features and changes in the latest TINA v7 version and also in the previous 6.0 and v5.5 of TINA. Many of them were suggested by TINA users, while others were created by DesignSoft's team of software and electrical engineers. We are sure you will share our excitement about these new features.

## 2.1 List of new features in TINA Design Suite 7.0

---

- Much faster analog solver algorithm with improved convergent properties
- Integrated VHDL support
- User defined VHDL components with VHDL source code
- VHDL components containing VHDL source code can be edited and executed instantly
- MCU support including wide range of PIC processors and more
- Built in debugger and assembler compiler for MCUs
- Assembler code of MCUs can be edited and executed instantly
- External simulator and debugger for VHDL
- 3D component view in the schematic editor to review if the design is PCB ready
- Passive and active filter design

- SMPS (Switching Mode Power Supply) design support (Steady State Solver)
- Control of interactive mode from the new Interactive menu
- Stress Analysis
- Advanced integrated PCB design
  - Multi-layer PCBs
  - Autoplacement
  - Autorouting
  - Rip-up and reroute
  - Follow-me trace placement
  - DRC
  - Forward and back annotation
  - Pin/Gate swapping
  - Keep-in/out areas
  - Thermal relief,
  - Fanout
  - Gerber file output
  - Copper pour
  - Split planes
  - Footprint editor with multi-pin footprint wizard
  - 3D view of PCB boards
- Support to design multi-pin schematic symbols
- More advanced Logic Design (simplification) tool
- IF statement allowed in Spice netlists
- More advanced file export (EMF, BMP, JPG) (File/Export)
- More advanced file import (EMF, WMF, BMP, JPG) (Insert/Graphics)

- Copy and Paste of any Windows dialogs (captured by Alt Prt Scr) into the Schematic Editor.
- Extended virtual instrument for real-time XY-recording, with average value, RMS calculation and recording vs. time

## 2.2 List of new features in TINA PRO 6

- Open several projects simultaneously and use copy and paste between projects
- Run analysis upon any open project
- New (.TSC) extension for project files to avoid collision with MS Schedule and ACCEL
- Store all analysis results (node and component voltages and currents) for post-processing
- Powerful post-processor for all analysis results
- XY-plot of any 2 transient or DC transfer results, for example  $v_1(t)$  vs.  $v_2(t)$
- Parameter stepping for any number of parameters
- Parameter stepping given by list
- Multiparameter optimization
- Draw lines, circles, arcs, and arrows in the schematic editor
- Add frame and title block to schematic diagrams
- Phasor diagram
- Voltage arrow to display voltages
- Diagonal lines for circuits such as bridges, delta, and wye (Y)
- Control switches with the keyboard via hotkeys
- Switch providing high and low logic levels directly
- Push button switches
- Fault insertion in wires
- Improved Library Manager for adding new models
- Improved method for changing component values in interactive mode with the mouse or with the keyboard via hotkeys
- Control the display of component values and labels separately

- Many more components: VCO, BSIM 3.3, communication circuits, 74121, 74122, and much more
- Component searching and listing tool
- Optional VHDL analysis
- Embedding faults into circuit files for advanced troubleshooting

### 2.3 List of new features in TINA PRO 5.5

---

- Fast 32 bit version for Windows 9x/ME/NT/2000
- Enhanced algorithms--now Spice 3F5 compatible
- 20,000+ built in components (5,000 previously)
- 1,000+ manufacturer-made components in Spice subcircuit format
- HTML-based component description
- New models including nonlinear coils, transformers, relays, darlington transistors, optocouplers, voltage regulators, fuses, thermistors and more
- Subcircuits which may include both Spice macros or schematics
- Automatic shape creation for subcircuits
- Schematic Symbol Editor for creating custom subcircuit components
- New enhanced wiring tool for easier circuit setup and modification
- Spice macros can be now used directly from TINA's schematic editor
- Parameter Extractor program to calculate model parameters from catalog or measurement data
- Spice Library manager for extending and maintaining manufacturer-made component libraries
- Multimeter, Oscilloscope and Signal Analyzer components
- Average and RMS value determination for arbitrary periodic waveforms
- Undo and Redo function with unlimited number of levels
- 64 Fourier components; both peak and RMS calculation
- AC Analysis; any number of sources, each with arbitrary frequency
- Component Toolbar Editor for adding your own components to TINA's component toolbar. Now you can create new component

# INSTALLATION AND START-UP

## 3.1 Installation Procedure

---

### 3.1.1 Minimum hardware and software requirements

- IBM PC AT/486DX or compatible (Pentium or better recommended)
- 64 MB of RAM
- A hard disk drive with at least 100 MB free space
- CD-ROM
- Mouse
- VGA adapter card and monitor
- Microsoft Windows 9x / ME / NT / 2000/ XP
- Novell Netware version 3.12 or later or MS Windows NT/ 2000/ XP Server or later for the Network versions

If the program is copy protected by a hardware key, the minimum hardware configuration includes also a parallel printer or USB port.

## 3.1.2 Installation from CD-ROM

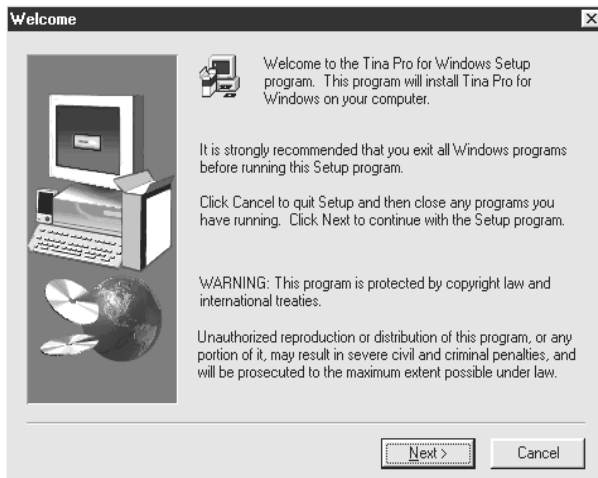
To begin the installation simply insert the CD into your CD-ROM drive. The Setup Program will start automatically if the Auto-Run function of your CD-ROM has been enabled (Windows-Default). If not, Select Start/Run and type:

D:SETUP (Enter) (where D represents your CD-ROM drive).




The setup program will start.

### NOTE:

This software may come with copy protection. For further details see the Copy Protection and the Network Installation sections.



### 3.1.3 Following the Installation Steps

TINA's Setup Procedure follows the steps standard with most Windows Programs. There are several screen pages where you can enter or change important installation choices, such as Type of Installation, Destination Directory, etc. To continue installation, click on . You can always step back, using the -Button. If you do not want to continue installation for any reason, click on . If you elect to cancel installation, the program will ask you if you really want to exit. At this point you can either resume or exit Setup.

### 3.1.4 Welcome and Software License Agreement

To begin the Procedure click on Next on the Welcome Page. The first step is the Software License Agreement.

**NOTE:**

By clicking on "Yes" you are agreeing fully with DesignSoft's Terms and Conditions for using this software.

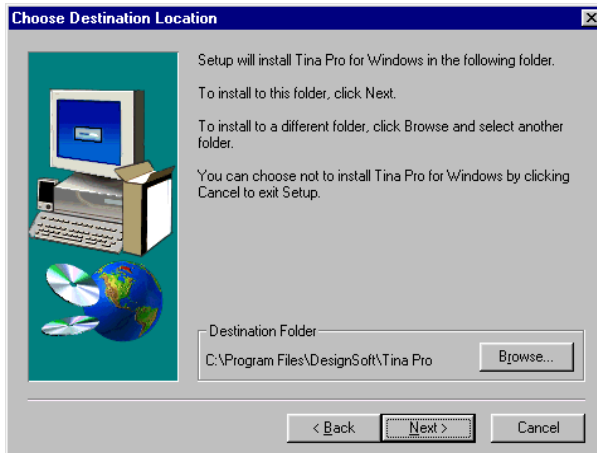
### 3.1.5 Entering User Information

This data is used to personalize your copy of the software. By default, the installation program picks up the data entered when you set up Windows. You accept these names as defaults by clicking on Next or you can change them.

Depending on your program version you might also need to enter a Serial Number located on your CD-ROM package or on your Quick Start Manual.

### 3.1.6 Choose Destination Location

Here you can select an Installation Directory other than the one suggested as a default. The default is the Windows Standard Directory for Programs. To change the directory, click on Browse and select a different drive and/or directory from the Choose Folder Dialog.

**IMPORTANT NOTE:**

If you are installing TINA for Windows to a hard disk that already has an earlier version of TINA, you must be sure to use a new directory name for TINA for Windows, such as the suggested directory, C:\Program Files\DesignSoft\Tina 7, or the working files you have already created will be overwritten and lost. If uncertain, exit setup, copy your TINA files safely to another hard disk directory or to floppy disks, then resume setup.



### 3.1.7 Selecting a Setup Type

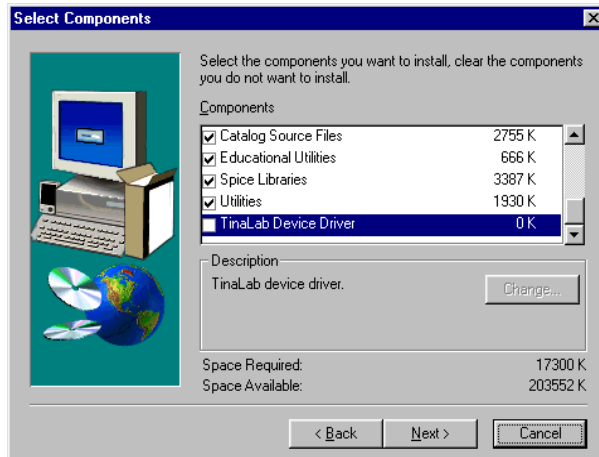
TINA offers you three different types of Setup. You can either run a Typical Setup (Default) a Compact Setup or a Custom Setup.

**NOTE:**

The detailed settings for the Compact installation are made after you select Compact and click on Next.

#### 3.1.7.1 Typical

The commonly used components are installed. This includes Program Files, Samples and Utilities (i.e. Exam Manager, Spice Library Manager).



### 3.1.7.2 Compact

Only the most important components are installed. This results in a usable TINA installation, but one without certain program components, such as Exam Manager.

### 3.1.7.3 Custom

You will be able to decide which components are to be installed. The default settings are similar to those of the Typical installation. Deselect the unwanted components or select the missing ones.

**NOTE:**

If you wish to install the TINALab Card, TINALab II or other third party supplementary hardware, you must select the Custom installation option at the time of installation and check the appropriate device driver on the list.

## 3.1.8 Selecting the Program Folder

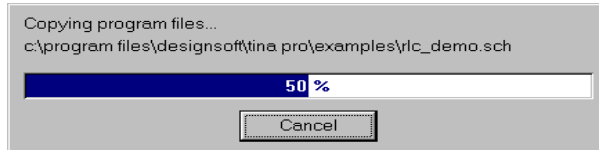
Here you can choose where in the Programs Section of your Windows Start Menu the Program Icons will appear. The default is a new sub menu called Tina 7. You can change this name or select an existing Program Folder from the list.

## 3.1.9 Selecting the Symbol Set

TINA can display its component schematic symbols according to the US (ANSI) or the European (DIN) conventions. Select the one appropriate for you.

### 3.1.10 Final check and copying the files

This page lists the settings you have made, giving you an opportunity to check and alter them and step back if changes are necessary. After you click on Next, the Setup Program starts copying the files automatically.



### 3.1.11 Completing the Setup

After all the selected files have been copied and the Start Menu entries created, you are asked if you want to place a Shortcut to the TINA program file on your Desktop. The last page indicates a successful installation and invites you to open and read a file with the latest information about TINA. We urge you to take a moment and review that file. Click on finish when you're ready.

**NOTE:**

You can read the latest information in the file again at any time by selecting Read Me from the Tina 7 Start Menu Entries. You can also get the latest information about changes or new features by visiting our Web Site, [www.tina.com](http://www.tina.com).

## 3.2 Uninstalling TINA

You can uninstall *TINA* at any time. Note that this will not delete files you have created.

1. To begin Uninstallation choose **Tina 7** from the **TINA 7 Start Menu** Entries.
2. In the Window that appears, double-click on **Uninstall Tina**.
3. Click on **Yes** if you are positive you want to uninstall *TINA*.

After all the files have been removed successfully, an **OK** Button appears. Click on it and uninstallation is complete.

## 3.3 Network Installation

To install the Network version of *TINA*, you must logon to your server machine as a user with administrative privileges (Novell 3.x: supervisor, Novell 4.x: admin, Windows NT: Administrator). Then execute the procedure for the Hard Disk Installation on a disk volume, that is accessible from the network. Now carry out the following additional steps:

Make all files in the program and user directories sharable:

### Novell 3.x:

```
FLAG *.* S SUB
```

### Novell 4.x:

```
FLAG *.* +SH /S
```

### Windows NT/2000/XP:

Depending on the setup of your system, you can give the rights to a group (<groupname>) whose members will then have the appropriate rights automatically.

Next make sure that the clients have a mapped drive set to the network drive containing the *TINA* program folder.

To assign (map) a drive letter to a network computer or folder do the following:

1. Open Windows Explorer
2. On the **Tools** menu, click **Map Network Drive**.
3. In **Drive**, select a drive letter e.g. **G:**
4. In **Path** (Win9x/Me/ or **Folder** (NT/2000/XP), select from the drop-down list or type in the network drive (server and share name) or folder name to which you want to assign (map) a **Drive** letter. Note, that share name refers to a shared folder on the server. On Windows NT/2000/XP you can use **Browse** to find the network computer, drive and folder.
5. Set the **Reconnect at Logon** checkbox.
6. Press **OK**
7. **Examples:**

Drive: **G:**

Folder: `\\servername\sharename`

or

`\\MyServer\Volume1`

`\\MyServer\Volume1\Public`

After you have set everything up on the Network disk according to the instructions above, you must run the setup program on each Novell workstation where you want to run *TINA*. Using the Run command, start **NSETUP** from the **Tina 7\NWSETUP** directory. Note, do not doubleclick on the NSETUP icon, but use the Run command instead to start the program.

When you run NSETUP you must specify the working directory which should be located on a local drive of the workstation. **The working directory can be on the network; however in this case the path of this directory must be different on every work station.** After you've specified the working directory, you may install the optional measurement hardware for *TINA* (e.g. *TINALab*). After running

*NSETUP*, you will be able to run *TINA* simultaneously on any number of workstations, just as though each workstation had a single user version.

### 3.4 Copy Protection

---

#### 3.4.1 Copy Protection by Software

If your version of *TINA* is copy-protected by software, at the first run the program, a message box appears showing your initial authorization status. You will normally have 31 similar, full featured sessions to provide you enough time obtain the authorization. If your program comes with an **Order number** the best it to press the Authorize button on the message box and enter your Order number into the Order number field of the Authorize dialog appearing and press OK. **Note that you must be on-line on the Internet when you enter your Order number.**

For more information refer to the program's Authorization Help by pressing the Help button, read the CD insert or the "Registration and License Control" manual.

In some cases the program comes with a **Serial number**, which needs to be entered during installation. When entering the serial number you do not need an active internet connection.

#### 3.4.2 Copy Protection by Hardware (dongle)

If you have a hardware-protected version, plug the dongle (hardware protection key) into the parallel (printer) or USB port connector. If you have a printer, connect it through this key. Should you forget to connect the dongle, an error message will come up on the screen:

Hardware protection key is not present

**NOTE:**

If you have a dongle protected version under NT/2000/XP you should install *TINA* in Administrator mode and restart the computer after installation.

## 3.5 Starting Up

After successfully installing *TINA*, you can start the program by simply double-clicking the *TINA* icon on your Desktop or by choosing *Tina* from the Tina 7 Start Menu Entries.

## 3.6 Experimenting with Example Circuits, avoiding common problems

Start the program and click the *File* menu item in the top line of the screen to drop down the *File* menu. Select the *Open* command and the standard open file dialog box appears with \*.*TSC*, indicating that a file name with .*TSC* extension is sought. Select the *EXAMPLES* folder, and a list of files with .*TSC* extensions will appear. After selecting a file, the circuit schematic will appear.

Now you can execute an analysis, modify or expand the circuit, and evaluate the results. Keep in mind that every command may be aborted by pressing the *[Esc]* key or clicking on the *Cancel* button.

We recommend that you load the following circuits and follow the instructions on the screen for the circuit types listed below. This will avoid some common problems.

<b>Oscillator circuit</b>	EXAMPLES/ <i>colpitts.tsc</i>
<b>555 Oscillator</b>	EXAMPLES/ <i>555_AST.tsc</i>
<b>Rectifier circuit</b>	EXAMPLES/ <i>Bridge Rectifier1.tsc</i>





## CHAPTER 4

# GETTING STARTED

In this chapter, we present TINA's screen format and menu structure. A step by step introduction is given using examples.

## 4.1 Schematic Editing Using the Mouse

---

Here are some basic mouse techniques to help you edit schematics:

### 4.1.1 Using the right mouse button

If you press the right button of the mouse at any time, a popup menu appears. Using this menu you can:

- **Cancel Mode:** Exit from the last operation (e.g. moving a component, drawing a wire).
- **Last Component:** Return to the last component and reposition it.
- **Wire:** Switch to wire-drawing mode. In this mode, the cursor turns into a pen and you can draw a wire. For more details, see the **Wire** paragraph below.
- **Delete:** Delete selected component(s).
- **Rotate Left, Rotate Right, Mirror:** Rotate or mirror the component currently selected or being moved. You can also rotate a selected component by pressing the Ctrl-L or Ctrl-R keys.

- **Properties:** Use this command to edit the properties (value, label) of the component currently selected or being moved. From the Properties menu, you can set all parameters of a component (before it is placed). This lets you place multiple copies of the component, all with the properties just entered. While you are in the component property editor, the right mouse button has another function. When you are editing the field of any component parameter other than the label field, you can copy that field to the label field by pressing the right mouse button and then selecting the *Copy to Label* command. You can accomplish the same thing by pressing [F9].

### 4.1.2 Using the left mouse button

In the descriptions below, ‘clicking’ always refers to the left mouse button.

- **Selection:** Clicking on an object will select the desired object and deselect all other objects.
- **Multiple selection:** Clicking while holding down the [Shift] key will add the object under the cursor to the group of currently selected objects. If the object under the cursor is already in the currently selected group, clicking will remove it from the group.
- **Block selection:** To select a block of objects all at once, first make sure there is no object under the cursor. Then press and hold down the left button while moving the mouse (dragging). This will create a rectangular block, and all objects within the block will be selected.
- **Selection of all objects:** Press Ctrl+A to select all objects.
- **Moving objects:** A single object can be moved by dragging it (Position the cursor on the object, press and hold the left button, and move the mouse.) Multiple objects can be moved by first selecting them (see above), then clicking the left button while the cursor is over one of the selected objects, holding the left button down, and dragging.

- **Parameter modification:** Double-clicking on an object will bring up its parameter menu so that you can modify its parameters (if it has any).
- **Crossing wires:** The crossing of two wires does not result in a connection at the crossing unless you deliberately choose there to be one. Use Edit.Hide/Reconnect to place or remove a connecting dot. However, it is better drafting practice to never make a connection at a wire crossing, as this avoids ambiguity about the presence or absence of a dot.
- **Block or symbol copying:** After a block or symbol has been selected, you may copy it by pressing Ctrl+C. Then click outside the block or symbol to release it, and press Ctrl+V. You will see a copy of the block which you can place as you wish. If the schematic window doesn't show enough room for the copy, press Alt - to zoom out. Once you've located the block, click the left mouse button once to anchor it and a second time to deselect the moved block.

## 4.2 Measurement Units

When setting parameters for electronic components or specifying numerical values, you may use standard electronic abbreviations. For example, you can enter 1k (ohm) for 1000 (ohm). The multiplier abbreviations should follow the numeric value, e.g., 2.7k, 3.0M, 1u, etc.

The following characters indicate multiplier factors:

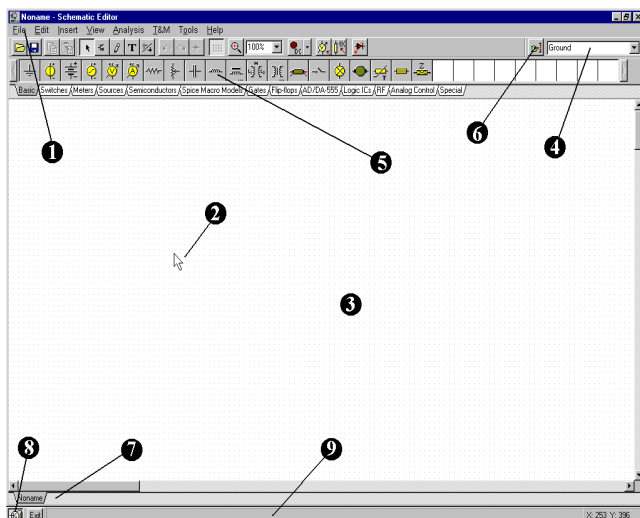
<b>p</b> = pico = $10^{-12}$	<b>T</b> = tera = $10^{12}$
<b>n</b> = nano = $10^{-9}$	<b>G</b> = giga = $10^9$
<b>u</b> = micro = $10^{-6}$	<b>M</b> = mega = $10^6$
<b>m</b> = milli = $10^{-3}$	<b>k</b> = kilo = $10^3$

**NOTE:**

Upper and lower cases must be carefully distinguished (e.g., M = m), and the selected letter must follow the numeric characters without a space (e.g., 1k or 5.1G), or TINA will indicate an error.

## 4.3 The Basic Screen Format

After start-up, the following screen appears on your monitor:



- ❶ **The Menu bar**
- ❷ **The Cursor or pointer:** This is used to select commands and to edit schematics. You can move the cursor only with the mouse.  
Depending on the mode of operation, the cursor assumes one of the following forms:  
 An **arrow**, when a command selection is required in the edit window.  
 A **component symbol** (accompanied by an arrow and small box), when inserting that component onto the circuit in the schematic window. Until the position of the component on the schematic is chosen, its movement is controlled by the mouse.  
 A **pen**, when defining the endpoint of a wire.  
 An **elastic line**, when defining the endpoint of a wire or the second node of an input or output.  
 An **elastic box**, when defining a block after fixing its first corner.  
 A **dashed line box**, when positioning a component label or a text block.  
 A **magnifying glass**, when defining a zoom window.

- ③ **The Schematic window:** This shows the circuit schematic currently being edited or analyzed. The schematic window is actually a window onto a larger drawing area. You can move the screen window over the full drawing area using the scroll bars at the right and bottom of the screen. When selecting the New command on the File menu, the system automatically aligns the origin of the editor window with the center of the entire editor drawing area. The same is true when an existing circuit file is loaded, as this is the default window position.

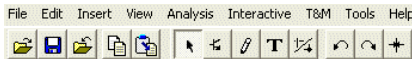
You can think of *TINA3* schematic as existing on several “layers.” In addition to the primary layer that holds components, wires, and text, there are two other drawing layers, which you can turn on or off individually. It is generally convenient to have these two layers on.


**View | Pin Markers On/Off:** Displays/hides component pin ends.

A grid of closely spaced dots covering the entire drawing area may be made visible or invisible in the schematic window, depending on the current state of the *grid button* of the *Grid On/Off* switch on the *View* menu. At some schematic zoom levels, you will not see the dots of the grid; nevertheless, all component pins and connecting wires will be on the grid. These dots represent the only available interconnecting points. Component symbols are positioned on the drawing area horizontally and vertically. These symbols are rigid patterns with predefined pin positions and are handled as single units. This permits the software to unambiguously recognize the network nodes.

**View | Grid On/Off:** Displays/hides the grid.

- ④ **The Tool Bar:** You can select most of the editor commands (e.g., select, zoom, wire etc.,) from this tool bar. Let’s summarize the most important commands on the Toolbar. You can find more detailed information in *TINA*’s Help system. Note that most commands on the toolbar can also be found under the drop down menus, and can often be activated by Hotkeys. We show the menu name separated from the command name by a dot (Menu name.Command name ).



 (*File.Open*) Opens a schematic circuit file (.TSC or .SCH) or macro (.TSM)

The .TSC is extension used in *TINA* v7, and v6, .SCH is used in earlier versions.

.TSM is the extension of TINA macros, which can contain a subcircuit either as a TINA schematic, Spice netlist, or VHDL code.



*(File.Save)* Saves the actual circuit or subcircuit into its original storage location. It is wise to frequently save the circuit that you are working on to avoid loss of data in case of a computer crash.



*(File.Close)* Closes the actual circuit. It is very useful for closing an open subcircuit.



*(Edit.Copy)* Copy a selected part of the circuit or text to the clipboard.



*(Edit.Paste)* Paste Clipboard contents into the schematic editor. Note that the content may come from the schematic editor itself, TINA's diagram window, or any other Windows program.



*Selection mode.* If this button is pressed you can select and drag components with the cursor. To select a component (part), wire or text, just click on it with the cursor. You can also select several objects by holding down the Shift key and clicking on the objects one-by-one; or by clicking at one corner of the area, holding down the left mouse button, moving to the opposite corner and then releasing the mouse button. Selected objects will turn red. You can drag the selected objects by dragging one of them. Click and hold the left mouse button when the cursor is over one of the selected objects and move them with the mouse. You can unselect all selected object(s) by clicking on an empty area. One or more selected objects can be deleted while leaving the others still selected by holding the Shift key down and left-clicking the mouse.



*(Insert.Last component)* Retrieves the last component inserted, for a new insertion of another copy, with the same parameters as the previous insertion.



*(Insert.Wire)* Use this icon for inserting (adding) wires to the schematic design.



*(Insert.Text)* Add comments into schematics and analysis results.

## Tool Bar Functions



(*Edit.Hide/Reconnect*) Use Hide/Reconnect to place or remove a connecting dot between crossing wire or wire-component connection.



(*Edit.Rotate Left (Ctrl L)*), (*Edit.Rotate Right (Ctrl R)*), Rotates the selected component.



(*Edit.Mirror*) Mirrors the selected component.

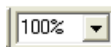
*Hotkeys* Ctrl L, Ctrl H



*Switches* On/Off the grid. I.e.makes the grid visible or invisible.



*Explicitly zoom* in on a selected portion of the current view. It will also zoom out a portion if you just click with the icon into the middle of the area you want to zoom out.



*Select zoom ratio* from a list from 10% to 200%. You can also select Zoom All which will zoom the effective drawing to full screen.

*Interactive modes*, see also on the Interactive menu:



DC mode



AC mode



Continuous transient mode



Single shot transient mode, the time is the same as set at Analysis Transient



Digital mode



VHDL mode



With this listbox you can select the analysis and the Interactive Mode Options dialog.





(*Analysis.Optimization Target*) Select Optimization Target to set up Optimization mode or to change settings.



(*Analysis.Control Object*) Select Control Object for Parameter Stepping or Optimization.



(*Analysis.Faults enabled*) If this button is pressed it enables component Faults, set by Fault property of components. You can set component faults with the Property Editor by double-clicking on components.



(*View.3D view/2D view*) Hotkey F6. 2D/3D view. If this button is pressed, TINA's Schematic Editor displays circuit components as 3D pictures of the real component assigned to the schematic symbol. This is a simple but useful check before starting the PCB design.



(*Tools.PCB Design*) Invokes the dialog which initiates the PCB design module of TINA.



(*Tools.Find component*) Find Component tool. Invokes a component searching and placement tool. This tool helps you find by name any component in the TINA catalog. The search string that you enter will be found wherever it occurs at the beginning, at the end, or anywhere within the component's name. This tool is useful when you don't know where a particular component is located, or if you want a list of all the components that match particular search criteria. A component found in a search can be placed immediately into the schematic by selecting it and pressing the Insert button of this tool.




(*Component list*). With this tool you can select components from a list.

- 5 **The Component bar:** Components are arranged in groups, named by the tabs on the Component bar. Once you have selected a group, the available component symbols appear above the tabs. When you click on the desired component (and release the button), the cursor changes to that component symbol and you can move it anywhere in the drawing area. You can also rotate the component by pressing the + or - keys (on your computer's numeric keypad) or mirror it by pressing the asterisk (\*) key (also on your computer's numeric keypad). Once you have selected the part's position and orientation, press the left button of the mouse to lock the symbol in place.
- 6 **Find component tool:** This tool helps you find by name any component in the TINA catalog. See more details above at the toolbar description.
- 7 **Open files tab:** You can have several different circuit files or different parts (macros) of a circuit open in the schematic editor at the same time. Clicking on a tab brings that circuit page up in the editor.
- 8 **The TINA Task bar:** *TINA's* Task bar appears at the bottom of the screen and provides speed buttons for the various tools or T&M instruments currently in use. Each tool or instrument operates in its own window and can be made active by clicking on its speed button (icon of the tool). Once the cursor is over the speed button, a brief hint appears. Note that the first button (furthest to the left), the Lock schematic button, has a special function. When the Lock schematic button is pressed, the schematic window is locked in place as a background behind other windows, so that it can never cover a diagram or virtual instruments. When the schematic window is not locked and it is currently selected, you will always see the entire schematic window with any other windows hidden behind.
- 9 **The Help line:** The Help line, at the bottom of the screen, provides short explanations of items pointed to by the cursor.

## 4.4 Placing the Circuit Components

Components are selected from the Component bar and their symbols are moved by the mouse to the required position. When you click the left mouse button, the program locks the pins of the component symbol to the nearest grid dots.

Components can be positioned vertically or horizontally and rotated by 90-degree steps in a clockwise direction by pressing the **[+]** or **[Ctrl-R]** keys, or in a counterclockwise direction with the **[-]** or **[Ctrl-L]** key. In addition, some components (like transistors) can also be mirrored around their vertical axis by using the **[\*/]** key on the numeric keypad. You can also use the  buttons or the popup menu (right mouse button) to position components.

After a component symbol has been selected and positioned, you may double click on it to enable a dialog window where you can enter parameter values and a label. When entering numeric values, abbreviations of integral powers from  $10^{-12}$  to  $10^{12}$  can be used. For example, 1k is understood as 1,000.

### NOTE:

Press the Help button on the component dialog box to invoke TINA's HTML-based help resource. You will find the parameters and mathematical model of the selected component. You can also access Component help from the help menu.

*TINA* will automatically assign a label for each component you place on the schematic. It will also display the numerical value of the main component parameter (for example: R4 10k). Note that the value is shown only if the Values option of the View menu is checked. For files from the older versions of *TINA*, the Values option is turned off by default. The first part of the label, e.g., R4, is required for symbolic analysis modes. You can also display the units of the capacitors and inductors (for example: C1 3nF) if both the Values and the Units options of the View menu are checked.

### 4.4.1 Wire

A wire establishes a simple short (zero ohm connection) between two component pins.

To place a wire, move the cursor to the component terminal point where you want to begin. The cursor will change into a drawing pen. Depending on View.Options settings you can draw a wire in two different ways:

- 1) Select the starting point of the wire with a left mouse click, then move the pen with the mouse while TINA draws the wire along the path. While drawing the wire, you can move in any direction and the wire follows. At the end point of the wire, click the left button of the mouse again. This drawing mode is the default setting in version 6 of TINA.
- 2) Hold down the left mouse button while positioning the pen; release it at the end point.

The actual wire-drawing mode depends on the editor options settings in View.Options. TINA by default uses the first method, but the program will remember that you have changed wire-drawing mode.

While drawing a wire, you can delete previous sections by moving backwards on the same track. By pressing the Ctrl key while drawing you can move the last horizontal or vertical section.

You can easily modify existing wires by selecting and dragging sections or edges.

For short wire sections, you may need to hold down the shift key while drawing.

You can also invoke the Wire-drawing tool by the Insert|Wire command (hotkey: [Space]). You can start drawing the wire at any place by clicking the left button of the mouse. When you have completed wiring, use the popup menu, press the right mouse button, or press the Esc key to terminate the wiring mode.

Be sure not to leave any component nodes unconnected. If there are unconnected components or terminals, TINA's Electric Rule Check tool (ERC) will issue a warning (unless you have disabled it).

Wire segments made by the Wire tool are always vertical or horizontal. However, you can add angled wire segments using the components made for bridges, Y and D circuits under the Special component toolbar.

## 4.4.2 Input and Output

Certain types of analysis (DC Transfer characteristic, Bode diagram, Nyquist diagram, Group delay, Transfer function) cannot be executed until both input and output have been selected. These establish where the excitation is applied and where the circuit response is taken. The output(s) chosen also determine which curve(s) will be displayed in the chosen analysis mode. Sources and generators can be configured as inputs, while meters can be configured as outputs. However, meters can also serve to determine the location of the input quantity that will be used when computing AC Transfer curves and functions. For even greater flexibility, inputs or outputs can be established at nearly any location by using the **Insert | Input** and **Insert | Output** commands. Note that you can define the input parameter for parameter sweeping only through the **Insert | Input** command.

To insert an input or output, select the Input or Output command from the Insert menu and move the input (**I+**) or output (**O+**) symbol attached to the cursor over the first schematic node that it should define. Click on that node, release the mouse button, and move the symbol to the second node, and click on that node. The program will draw a dashed rubber line between the two nodes while drawing, and will also place this line on the schematic when you click on the second node.

Since an input reference can be established in so many ways, it is important to remember that only one input at a time can be defined within a circuit.

Similarly, in some of *TINA's* analysis methods (e.g., Symbolic Analysis) only one output can be defined within a circuit.

## 4.5 Exercises

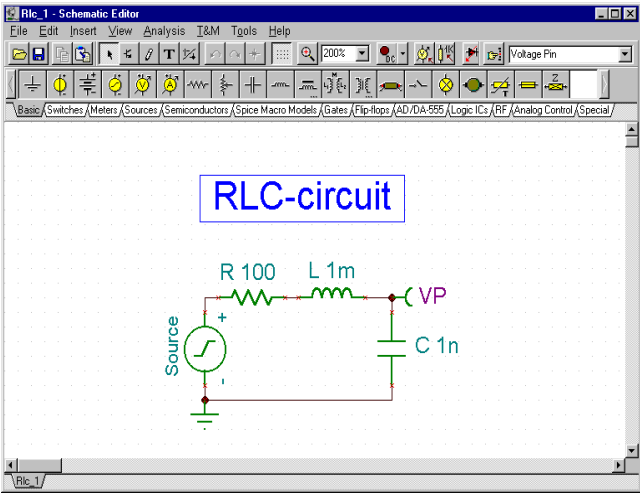
These exercises will help you build upon and integrate what you’ve learned from the manual so far.

### 4.5.1 Editing an RLC Circuit Schematic

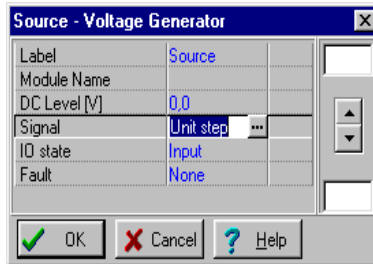
Create the circuit diagram of a series RLC network as shown in the following figure.


First clear the schematic window with the **File | New** command. The file name in the top line is set to Noname, indicating that a new circuit file is being edited.

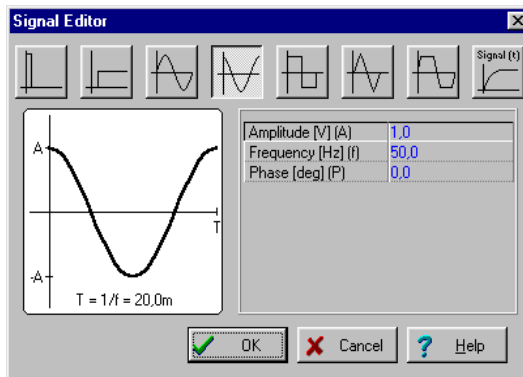
Now start adding components. Click on the voltage generator icon, then release the mouse button. The cursor will change into the generator symbol. Position it using the mouse (or by pressing the **[+]/[Ctrl-R]** or **[-]/[Ctrl-L]** key for rotation) or the **[\*]** key for mirroring) somewhere in the middle of the screen, then press the right mouse button; the schematic editor’s popup menu will come up. Select *Properties*. The following dialog box will appear:



Leave the *DC level* and the *IO state* parameters unchanged. Note that by accepting *Input* for the *IO state* parameter you have selected the output of this generator to be the input for the Bode diagram.

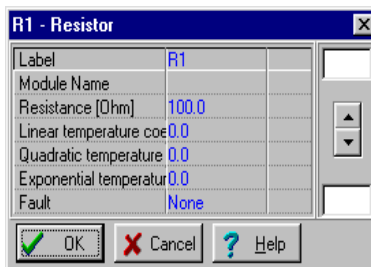


Select the *Signal* menu line and then press the  button, a new dialog box with the graphics icons of available voltage generator signals will appear. When you select one of them (in this case, click on the Cosinusoidal button), the associated curve comes up with some default parameters. In the case of the Cosine signal, these are:



Change the frequency to 200k (200kHz). Click on *OK* and return to the previous dialog box and click on *OK* again. The program will automatically place the label near the component and you will be able to position and place the component and the label together. If the default label position is not satisfactory, you'll be able to drag the label to the desired position later on. When the component is where you want it, press the left mouse button to drop it. This completes the placement of the generator.

Now click on the *Basic* tab on the Component bar and choose the **Resistor** icon (your cursor will automatically change when you are over the tabs or the icons). After the symbol of a resistor has appeared in the schematic window, press the right button of the mouse and then select *Properties* from the popup menu. When the following dialog box appears, change the Resistance to 100:






After setting all parameters, click *OK*. Your cursor will turn into the resistor with the frame of the label. Position it as required and press the left mouse button to drop it.

Continue circuit entry with the **L** and **C** components as indicated in the figure above. Set the parameters to  $L = 1 \text{ m}$  and  $C = 1 \text{ n}$ . Note the default values of the parallel resistive losses for the capacitor and the series resistive losses of the capacitor. Add the Voltage Pin (chosen from the Meters component group) on the upper pin of the capacitor (or you can add a volt- meter in parallel with the capacitor). Note that even though all the computed voltages, currents and signals are available after running an analysis (see below in this chapter and also in the *Post-processing analysis results* section), you still need to define at least one output. Place a ground below the generator and connect the generator and capacitor as shown in the figure. To do this, move the



cursor over the appropriate pin node until the small drawing pen appears. When the pen appears, click the left button of the mouse, draw the wire, and left click again at its end point.

Finally, add the title to the schematic. Click the  button and the text editor will appear. Type in: RLC Circuit. Click on the  icon and set size 12. The editor also lets you choose another font, style, color etc. Now click on , then position and drop the text on the schematic editor Window.

Before going on, save the circuit with the *File|Save As* command. Name the circuit as **RLC\_NEW.TSC** (the .TSC extension is added automatically).

If you wish, you can still change the circuit in many ways:

- Add new components.
- Delete, copy, or move selected objects using the *Edit|Cut, Copy, Paste, and Delete* commands.
- Move, rotate, or mirror groups of components. Select the components one by one, holding down the shift key as you click on them. You may also use window selection to identify the group. When you've selected the last component, release the left mouse button, then move the cursor over one of the selected components, press and hold down the left mouse button, and drag the selected parts with the mouse. While dragging, you can use the **[+]/[Ctrl-L]/[Ctrl-R]**, **[-]** and **[\*]** keys to rotate and mirror components.
- Move any component label separately by clicking on it and dragging.
- Modify component parameter value(s) and labels of the component by double-clicking on it.

Of course, if you want to keep these changes you must save the circuit again.

## 4.6 Analyses

*TINA* has a variety of analysis modes and options:

The analysis method is analog when a circuit contains only analog components; then the components are modeled with their analog models.

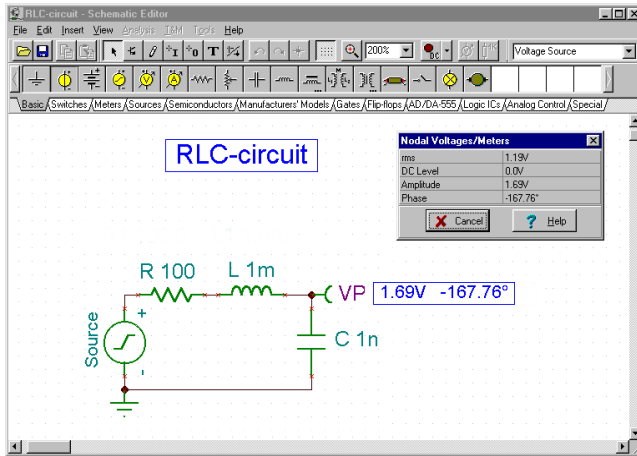
The analysis method is mixed when a circuit contains both analog and digital components; then the components are modeled with their analog models.

The analysis method is digital mode when a circuit contains only digital components; then the components are modeled with their fast digital models.

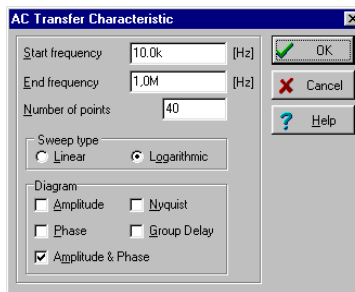
### 4.6.1 Analyzing an RLC circuit ( DC, AC, Transient and Fourier analysis )

Now execute AC and transient analyses on the RLC circuit you have just entered.

First perform an AC nodal analysis. Select **Analysis | AC analysis | Calculate nodal voltages**. Your cursor will turn into a test probe which you can connect to any node. In a separate window the nodal voltages will be displayed. If you have placed any meters on the schematic, clicking on them with the probe will present detailed information from that instrument. Note that you can acquire DC nodal voltages in a similar fashion through DC Analysis.

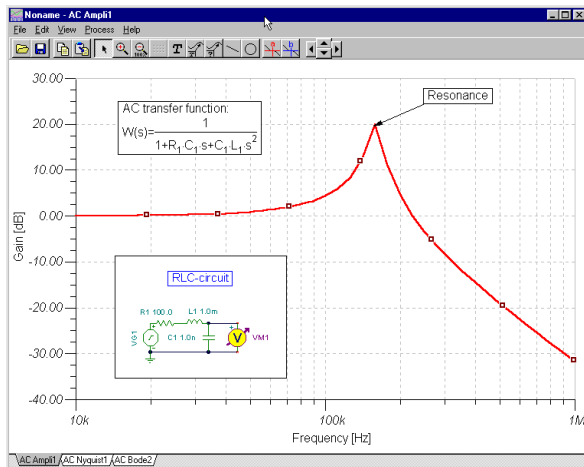


Now select **AC Analysis | AC Transfer Characteristic...** from the main menu. The following dialog box appears:



By default Amplitude & Phase will be calculated. Select **Amplitude** and **Nyquist** in addition. Modify the Start frequency to 10k and then press OK. A progress bar will appear while the program is calculating. After the calculations are finished, the Bode amplitude characteristic will appear in the Diagram Window. You can easily switch to Nyquist or Amplitude & Phase diagrams by using the Tabs at the bottom of the Diagram Window.

You can read exact input/output values by enabling one or more of the cursors. Note that in any representation you can get and place the formula of the transfer function using Symbolic Analysis and selecting AC Transfer or Semi Symbolic AC Transfer. The formula will appear in the Equation Editor Window and you can place it either on the Diagram or the Schematic window as described above.



Using *TINA*'s graphic facilities, you can add more useful information to your diagram. As an example, let's add markers, a special annotation, and the circuit schematic itself to the diagram.

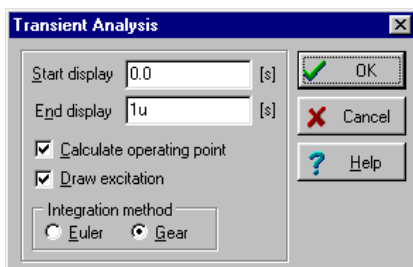
To add markers to a curve move the cursor over the curve, find a position when the cursor changes into a + shape and click on the curve at this position. When the curve becomes selected, it turns red. Now you can either double-click on it or press the right button of the mouse and select Properties on the popup menu. A dialog box will appear and you can set the curve's parameters: Color, Line width, Marker. Select Marker Type: Square and click on OK.

To add some text, click on the **T** text icon. When the Text editor appears, type in **Resonance**. Note that using the **F** font icon of the editor you can select any font, style, size and color. Click on OK and place the text in the neighborhood of the resonance peak. Now click on the pointer icon, then on the text, and finally on the peak of the curve. Note that the cursor turns into a + when you are at the right position. You have just entered a line and arrow that will always point from the text to the curve, even if you drag the text into another position or make other changes.

Now place the schematic itself on your diagram. Click the schematic editor Window and select *Edit | Select All*. Copy this selection into the clipboard by selecting *Edit | Copy* or clicking on the Copy icon or using the Ctrl C hotkey. Click on the Diagram Window and use *Edit | Paste*,

or click on the Paste icon, or use the Ctrl V hotkey. The frame of the circuit diagram will appear. Position and drop it at the left corner of your diagram. Now you can still modify this picture by dragging or double-clicking on it and changing its size, frame or background.

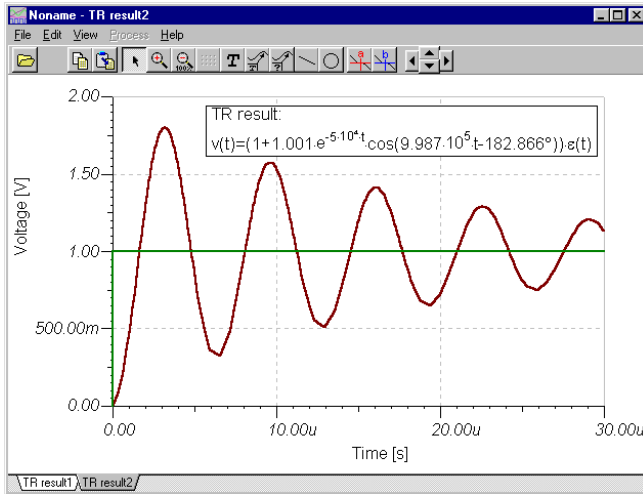
Now perform a transient analysis. First, make sure your cursor is the selection arrow, then double-click on the voltage generator and change the waveform to the default unit step. After selecting **Analysis | Transient Analysis**, the following dialog box appears:




Change the *End Display* parameter to 30 u then press OK. In a separate window the transient response will appear.

As expected, the RLC circuit exhibits a response of damped oscillation. Exact input/output data pairs can be read by enabling the a and/or b graphic cursors.

Now select **Analysis | Symbolic** or **Analysis | Semi-symbolic Transient** from the menu. The closed form expression of the circuit response appears in the Equation Editor window. Click on the Copy icon of the Equation Editor Window, then switch to the Schematic window and select the Paste icon. The frame of the formula will appear. Move the frame to the desired location and press the left mouse button to place the formula. Note that you can reposition it by dragging to any position and you can edit it by double-clicking on it.

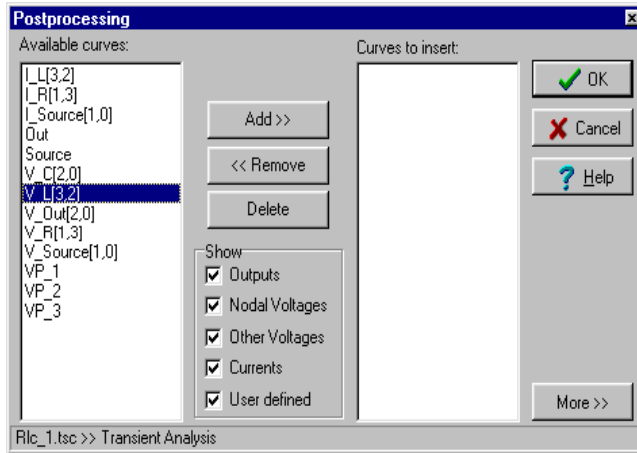


Now go back to the equation editor and click on the Interpreter icon (small calculator) on the toolbar. The expression shown in the equation editor is then transferred to the Interpreter window. The actual definition of the time function is at the top of the window, followed by drawing preferences and the 'Draw' command. Press the run button to draw the function in the diagram window on a new page. This curve can then be copied and pasted into the same transient function diagram, where all of the results can be seen simultaneously.

To add more analysis curves to our diagram, press the *Add more curves* button  on the toolbar or call the Add more curves command in the Edit menu. The **post-processing** dialog appears on the screen.

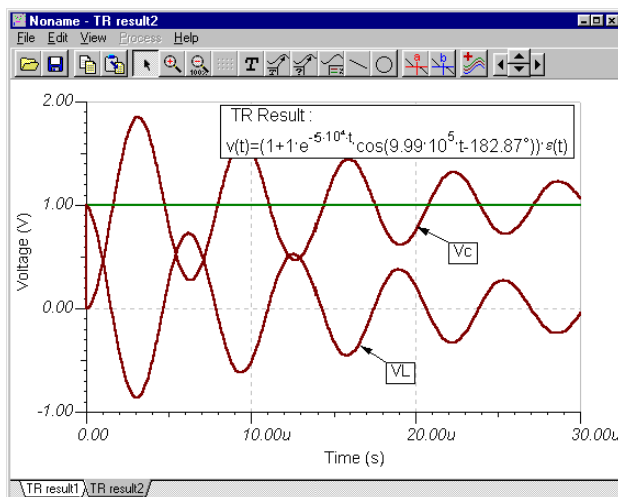
Note; in order to use this feature, you must check "Save all analysis results" in the Analysis.Options dialog

The available curves listbox on the left side shows all the curves that have already been calculated.




The symbols  $V_{\text{label}[i,j]}$  and  $I_{\text{label}[i,j]}$  denote the voltage and current respectively of the labeled components between the nodes  $i$  and  $j$ . The symbol  $VP_n$  denotes the nodal voltage of node  $n$ .

To add the voltage of the coil to the list of curves to insert, select  $V_L[3,2]$  and press the Add>> button. Pressing OK inserts this curve into the current diagram page.

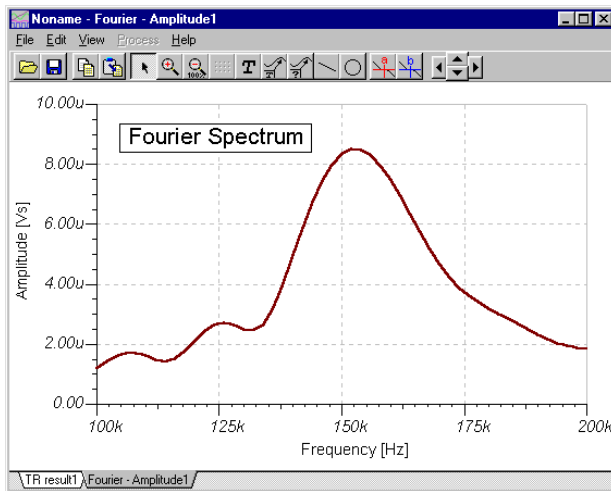


You can do a lot more with TINA's post-processing tool. For example, you can create curves of new functions created by adding or subtracting curves, or by applying mathematical functions on them. For a more detailed description, refer to the Post-processing analysis results section in the Advanced Topics chapter.

To demonstrate a more advanced feature of *TINA*, examine the *Fourier Spectrum* of the nonperiodic transient response just obtained. First, select the damping output signal by moving the cursor over the curve and pressing the left mouse button when the cursor assumes a + form. The selected curve will change to red. Now press the right mouse button and select *Fourier Spectrum* from the popup menu. The Fourier Spectrum dialog box will appear. Set the **Maximum frequency** to 200k and press OK. The Fourier spectrum of the transient response will appear. (You will get a finer curve if you repeat the transient analysis with 500us End Display value). Using the  zoom icon zoom into the spectrum between 100kHz and 200kHz. Note that you can also change settings by double-clicking on the coordinate axes. The result is a continuous frequency spectrum shown in the figure below. As expected, the frequency where the Fourier Spectrum shows a maximum is the resonant frequency of the circuit.

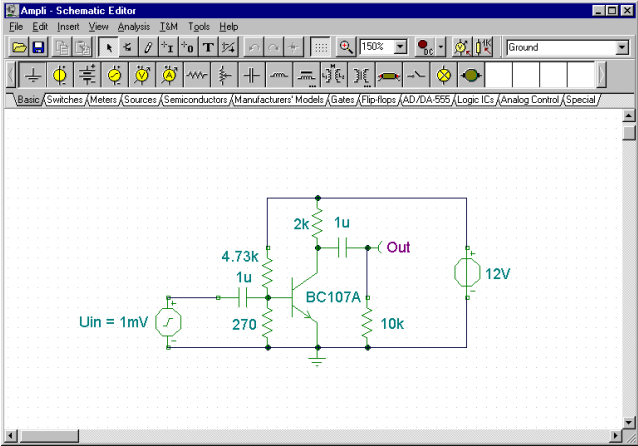
The Fourier spectrum and the Fourier series dialog box can also be obtained directly from the Analysis.Fourier Analysis menu. This way you do not need to calculate the transient function manually—TINA will automatically do it before generating the Fourier series or spectrum.



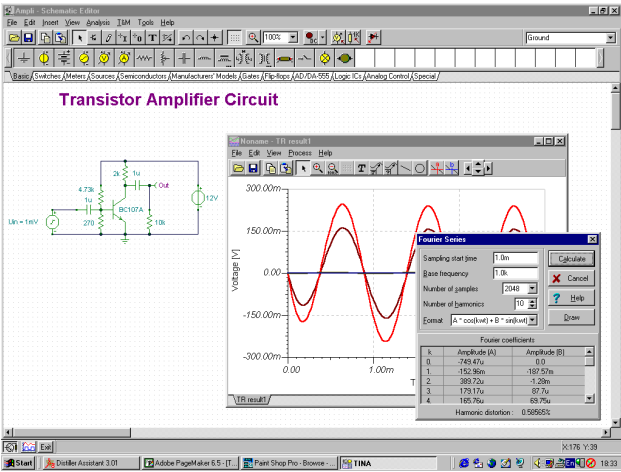
**NOTE:**

You might be surprised that the unit of the frequency spectrum is in Vs (volts times seconds). That is because the continuous Fourier spectrum is a density function versus frequency. If you want to know the approximate amplitude in a narrow frequency band, you multiply the average amplitude (given in [Vs]) in the band with the bandwidth (given in [1/s]).

Fortunately, Fourier analysis is not so complicated for periodic signals. Periodic signals can be represented by Fourier Series or in other words as a sum of cosine and sine waves at the base (fundamental) frequency and integer multiples of the base frequency. To try out this kind of Fourier analysis in *TINA*, load **AMPLI.TSC** from the *EXAMPLES* folder.



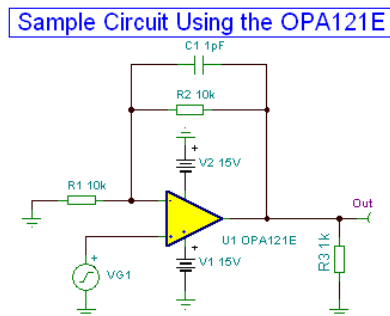
Run a transient analysis and then select the output curve with the largest amplitude. Press the mouse right button and select **Fourier Series** from the popup menu: the dialog box of the Fourier series will appear. Note that you can access this dialog directly from the Analysis. Fourier Analysis menu. Set Sampling start time to 1ms and the Number of samples to 2048. Note that for best accuracy, it is very important to set the starting time for the Fourier Series analysis to after the initial transient has died away. Now press *Calculate*. The list of Fourier components will appear.



If you press *Draw* you can also draw a diagram showing the amplitudes in V (volts) at integer multiples of the base frequency.

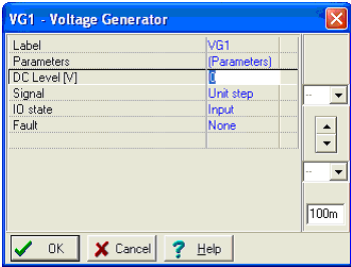
## 4.6.2 Creating and editing an analyzing an Op-Amp circuit

Create the circuit diagram using an OPA121E operational amplifier from Texas Instruments as shown in the following figure:

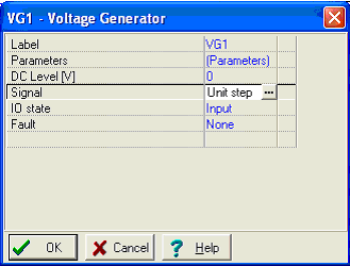



If you just opened TINA and wish to create your own circuit, you can start adding components right away. The circuit file name in the top line is set to Noname by default, indicating that a new circuit file Noname.TSC is being edited. If you already have a circuit loaded in the editor, for example, our previous RLC circuit, you can start a new one with the File|New command. You can switch between multiple circuits by clicking the Tabs at the bottom of the screen.

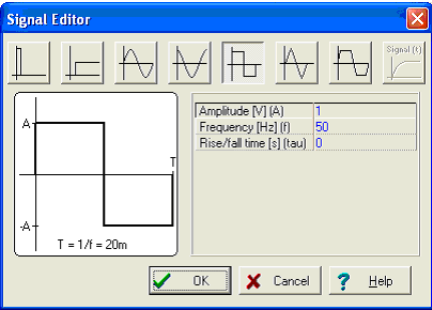
Now start adding components. Left click on the voltage generator icon then release the mouse button. The cursor will change into the generator symbol. Position it using the mouse (or by pressing the [+)/[Ctrl-R] or [-]/[Ctrl-L] key for rotation] or the [\*] key for mirroring) somewhere in the middle of the screen, then press the left mouse button to drop the component into the schematic. We still need to set the properties of this generator. Double-click on the generator and the following dialog box will appear:



Leave the DC level, and the IO state parameters unchanged. Note that by accepting Input for the IO state parameter you have selected the output of this generator to be the input for this analysis (a Bode diagram in this example). Click on the Signal menu line. The dialog box will change as shown below:

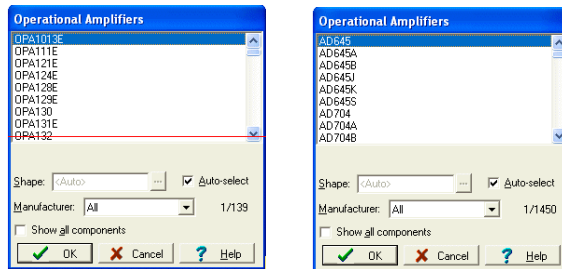


Press the  button. A new dialog box with the graphic icons of available voltage generator signals will appear. When you select one of them (in our case, click on the Square Wave button), the associated curve comes up with some default parameters. In the case of the Square Wave signal, these are:




Change the Amplitude to 500m (this represents 500mV peak), the frequency to 100k (100kHz), and the Rise/Fall time to 1p (1ps). Click on OK and return to the previous dialog box and click on OK again. The program will automatically place the label (VG1) near the component and you will be able to position and place the component and the label together. If the default label position is not satisfactory, you'll be able to drag the label to the desired position later on.

Now click on the Spice Macros tab and press the leftmost Operational Amplifiers button. The following dialog box will appear:

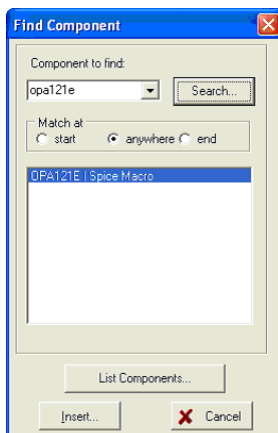


To find the IC we want, scroll down the list until you find OPA121E. You can narrow the list if you select the manufacturer (Texas Instrument in our example) from the Manufacturer listbox. You can also simply type OPA121E and the list will automatically jump to the IC (press the Delete button on the keyboard and try again if you make a typing error). Click on the line (OPA121E) and press the OK button. (Alternatively you can double-click on that line). The schematic symbol of this opamp will appear and be attached to the cursor. By moving the mouse, position the opamp as shown on the schematic at the beginning of the section and then press the left mouse button to place the opamp into your schematic.

You can also select a part using the  Find Component tool at the top-right corner of the Schematic Editor. If you type the part number into the "Component to find" field and press the Search button, the list of available component(s) will appear. (You can enter just part of the name if you are not sure of the entire name).

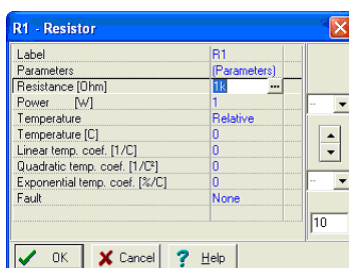
Press the Insert button to place the component. With the List Component button you can create the list of all available components in a textfile.

## Creating an Op-Amp circuit



Note that other types of ICs are available under the buttons next to the Operational Amplifiers: Difference Amplifiers, Fully-Differential Amplifiers, Comparators, Voltage Regulators, Buffers, Current Shunt Monitors, and Other Components). You can bring all of these various components into the dialog box for any of the buttons if you set the Show All Components checkbox. In addition to selecting an IC on the list, you can also find it by clicking on any item on the list and then typing in the name of the IC.



Now click on the Basic tab on the Component bar and click the Resistor icon. The resistor symbol will be attached to the cursor. Move the resistor to the position of the R1 resistor on the sample schematic diagram at the beginning of this section and press the left mouse button to place this resistor into the schematic. Double-click on the resistor and the following dialog box will appear:



Change the value in the Resistance field to 10k, and press OK.


Note that you can set a component value before placement while you are moving it. To do this, press the right button of the mouse and select Properties on the popup menu. The dialog shown above will appear and you can set the properties of that component. After pressing OK, you can return to component placement.

Now let's place R2 at the top of the circuit. Click on the resistor symbol on the component toolbar, move and place the resistor. When you drop the resistor you will see that its value is already 10k since the program remembers the previous value.

Now let's place R3, which needs to be turned by 90 degrees. Click on the resistor symbol on the toolbar and turn the component by 90° by clicking the  or  buttons or pushing the Ctrl L or Ctrl R keys. (+ and – on the numeric keyboard have the same effect). Place the component on the right side of the screen and set its value to 1k.

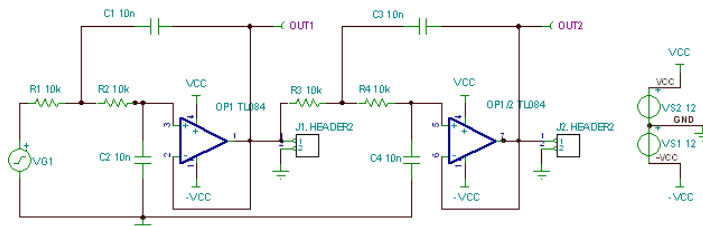
Continue circuit entry with the Capacitor, Battery, and Ground components as indicated in the figure above. Set the parameters to C=1 p, V1=15, and V2=15. Place a Voltage Pin Out (chosen from the Meters component group) at the right side of the new schematic. Pay attention to battery polarities and rotate the symbols if necessary.

## NOTE:

If you have several opamps you may want to simplify their connection to the power supply. This can be done using the  Jumper component, which you can find at the first place on the Special toolbar.

All jumpers with the same label are considered as electrically connected components in TINA. Therefore, if you connect a jumper called VCC with the positive power supply of the opamp, it is enough to connect jumpers with the same VCC label to the positive power supply pins of the opamps. As an example, you can load and study OPAMP2.TSC from TINA's EXAMPLES\PCB folder, also shown below.

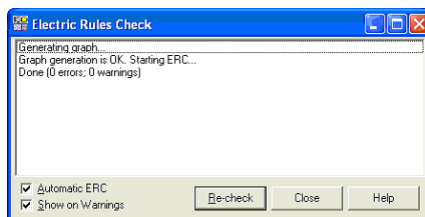
## Creating an Op-Amp circuit



Note that even though all the computed voltages, currents and signals are available after running an analysis (see below in this chapter and also in the Post-processing analysis results section), you still need to define at least one output. We have placed the parts into the schematic but they are still unconnected. To connect devices, move the cursor over an appropriate pin node until a small drawing pen icon appears. When this pen appears, click the left button of the mouse, draw the wire, and left click again at its endpoint.

Finally, add the title to the schematic, using the **T** icon on the toolbar.

Let's check the circuit we have just built and run ERC from the Analysis menu. If everything is OK, the following dialog will appear:

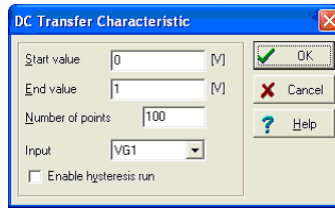


If there is a problem with the circuit, a list of warnings or error messages will appear in the dialog box. If you click on a warning or error message, the related part or wire will be highlighted in the circuit diagram.

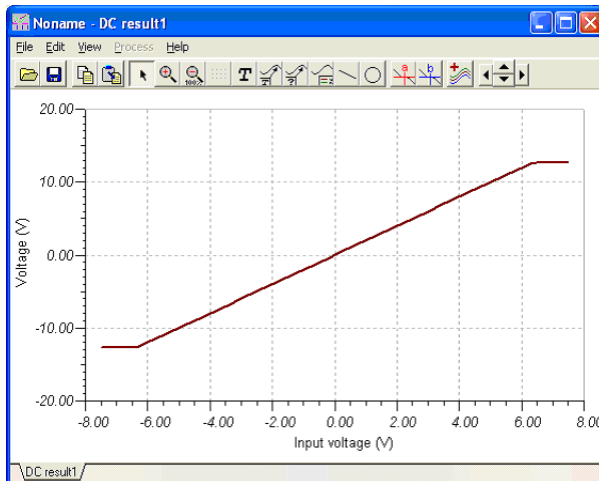
### 4.6.2.1 Calculationg DC Transfer characteristic

We have already seen several of TINA's analysis modes. But so far we have not used the DC analysis mode to calculate the DC transfer characteristic of this circuit. Select DC Analysis|DC Transfer Characteristic... from the Analysis menu. The following dialog box will appear:





Set the Start value to  $-7.5$ , the End value to  $7.5$ , and then press OK. After a short running time, a Diagram Window will appear as shown below. This displays the circuit's transfer curve-output voltage vs. input voltage.



### 4.6.3 Analysis of SMPS circuits

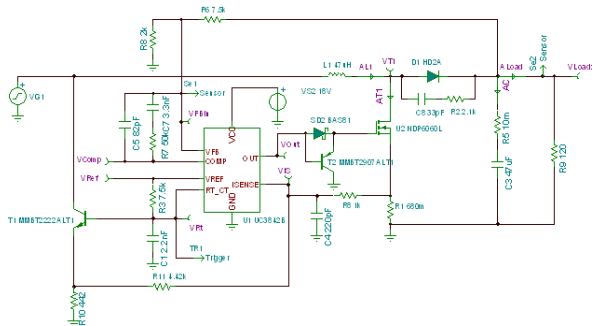
SMPS or Switching-Mode Power Supply circuits are an important part of modern electronics. The heavy transient analysis needed to simulate such a circuit may take a lot of time and computer storage. In order to support the analysis of such circuits TINA provides powerful tools and analysis modes. In this Chapter we will demonstrate these through examples.

Using the Steady State Solver

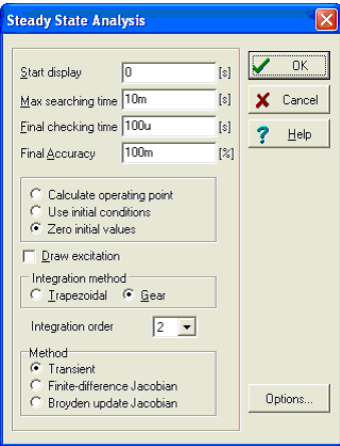
The most time consuming part of an analysis of an SMPS circuit is to reach its steady state, when the DC level of the output voltage does not change and the output waveform has only a small periodic ripple.

To find this state automatically, TINA has a Steady State Solver under the Analysis menu.

To demonstrate this tool, let's Load the UC3842 Boost Converter 5\_12V Steady.State.TSC circuit file from TINA's EXAMPLES/SMPS folder.



Select the Steady State Solver from the Analysis menu. The following dialog will appear:



The new parameters compared to the Transient Analysis dialog box areas follows -

*Max searching time:* The solver will try to find the steady state solution for max 10ms. After this, the analysis will discontinued whether or not a solution was found..

*Final checking time:* After the steady state search is done, there is a final check for the length specified here. You should have a stationary waveform for this time interval.

*Final accuracy:* the maximum DC level change allowed. When the change is below this, the analysis will end. Note that the 100m in the example above means 0.1%

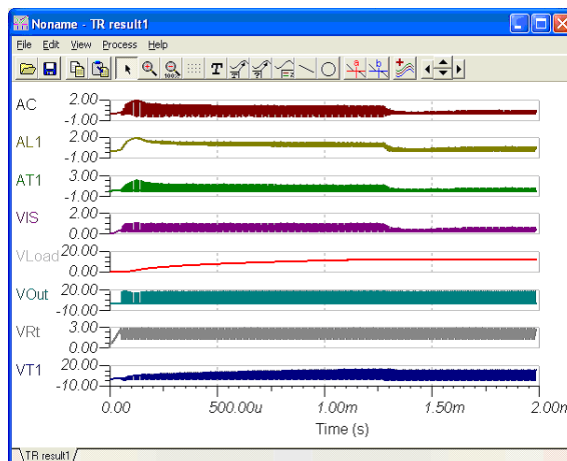
*Method:* You can select the method used for finding the steady state:

*Transient:* The steady state is searched by using transient analysis.

*Finite-Difference Jacobian, Broyden update Jacobian:* The steady state is searched by the methods described in the paper, *Automated steady-state analysis of switching power converters* by Dragan Maksimovic.

Note that these last two methods may get to the steady state faster, but they do not go through the normal transient states, so the resulting waveform between the initial and final state does not reflect the real process (but rather the mathematical path of the methods to get there).

Now lets run the Solver. After a few minutes of running (approximately 3 minutes on a 3GHz Pentium computer) we'll get the following resulting waveform:



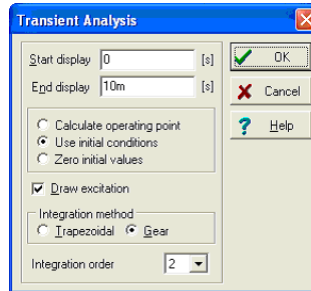
These waveforms show the detailed transient from switching on until reaching the steady output voltage. If you zoom out on the waveform, you can see that the period of the switching is around 100kHz and the time needed to arrive at the steady state is 2 milliseconds. Therefore, we need to calculate at least hundreds or sometimes thousands of periods if we want to see the whole transient waveform. This is why finding the steady state is such a time consuming process. The reason for this problem is the long start-up time of SMPS circuit compared with their switching frequency. The start-up time is basically determined by the filter capacitors on the output. The larger these capacitors are, the longer the start-up time.

**NOTE:**

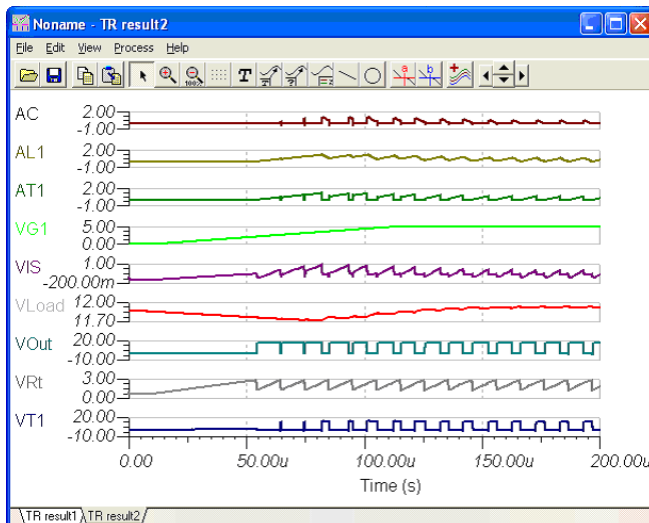
In some cases you can accelerate the Steady State Search using the Finite-Difference Jacobian and Broyden update Jacobian methods, however they do not always converge and the intermediate waveforms provided by these methods do not reflect the real waveforms of the transient process.

### Accelerating SMPS simulation using initial values

As we mentioned in the previous section, the long analysis time needed for reaching the steady state of SMPS circuits is mostly used for charging the output filter and some capacitors. If we start the analysis using initial values for larger capacitors and inductors, the analysis time can be significantly reduced. In TINA, the Steady State Solver will automatically place initial values into the model of larger capacitors and inductors and so the following Transient Analysis can be run significantly faster (assuming that we do not make changes which will need significantly different initial values). For example, if you want to study the effect of changing the output filter capacitor, it will not significantly change the output voltage DC level. Therefore, starting the new analysis with an initial value calculated by the steady state solver for another output capacitor, will result in a much faster analysis. You can accelerate the analysis of input and load changes in the same way.

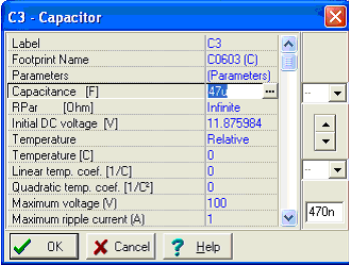


To demonstrate this feature, let's run a transient analysis for our example. Selecting the Transient command from the Analysis menu, the following dialog box appears.



Note that Use initial conditions is set in the dialog. Press OK to start Transient analysis. You should see that analysis runs very fast compared to the previous steady state analysis. The output waveform is shown in the picture below.

Why did the analysis run faster? Transient analysis was already preceded by the Steady State Analysis and the main capacitor's initial values (called Initial DC voltage in the Capacitor property box) was already set to the final DC voltage. For example, if you double-click on the C3 capacitor, you will see that the Initial DC voltage is already set to 11.875 V. Similarly, all the larger capacitor's initial values are set.

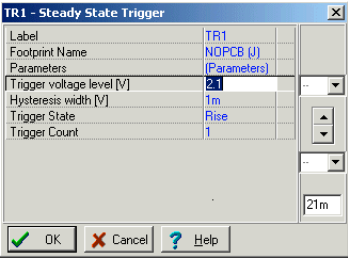


### Trigger

Use this to determine the starting and ending times of the switching period.

You can find this component on the Meters toolbar of TINA. You should connect it to the oscillator frequency control pin of the SMPS/ PWM controller IC, but any node where the oscillator waveform of the IC is present will do.

If you double-click on the Trigger component you can set its parameters.



*Trigger voltage level:* the threshold voltage for the trigger event

*Hysteresis width:* hysteresis value for the trigger event. This value defines a region within which the trigger voltage is allowed to oscillate without generating a trigger event.

*Trigger State:* Rise/Fall The direction of the voltage change required for a trigger event

*Trigger Count:* you can take several periods for the waveform analysis.

This is useful in case of very slowly changing output signals. The only electric parameter of the Sensor component is -

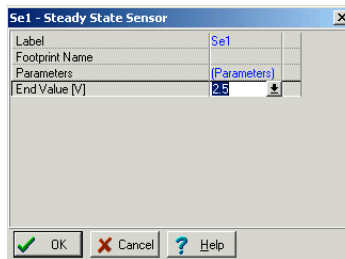
*End Value:* Voltage | Not Used

Using the “Max. no. of saved TR. points” parameter in the Analysis/Analysis parameters dialog, you can limit the maximum number of points placed in the Diagram. This is useful for large analyses to accelerate diagram drawing. By increasing this parameter, you can refine the diagrams but the drawing time will be slower.

Once you have checked the initial transient and the steady state waveform and SMPS circuit the next thing you normally want to know is how it behaves when the input voltage or the load changes. This is realized by the **Input step** and the **Load step** analyses.

### Sensor

The purpose of this component is to set the target voltage(s) to be watched during the steady state search. You can add more than one sensor to a circuit. By adding sensors, you can significantly accelerate the steady state search. You can make the search even faster if you can give the final voltage at a certain node.



Using the “Max. no. of saved TR. points” parameter in the Analysis/Analysis parameters dialog, you can limit the maximum number of points placed in the Diagram. This is useful for large analyses to accelerate diagram drawing. By increasing this parameter, you can refine the diagrams but the drawing time will be greater.

The only electric parameter of the Sensor component is -

*End Value:* Voltage | Not Used

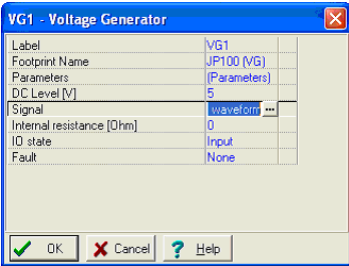
Once you have checked the initial transient, the steady state waveform, and SMPS circuit, the next thing you normally want to know is how it behaves when the input voltage or the load changes. This is realized by the **Input step** and the **Load step** analyses.

NOTE:

Using the “Max. no. of saved TR. points” parameter in the Analysis/Analysis parameters dialog you can limit the maximum number of point placed in the Diagram. This is useful for large analyses to accelerate diagram drawing. By increasing this parameter you can refine the diagrams but the drawing time will be slower.

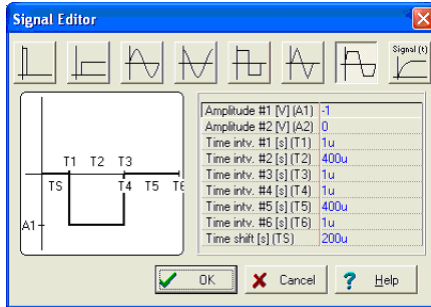
Input step analysis

One of the standard analyses for SMPS circuits is the calculation of the response to an input change to test the capability of the SMPS design to regulate the output with step changes in the input line. This can be accomplished by adding a pulse to the input voltage and checking the output and other voltages. Since the input change is relative to the steady state, we can start it from the steady state initial values calculated by TINA’s steady state solver.




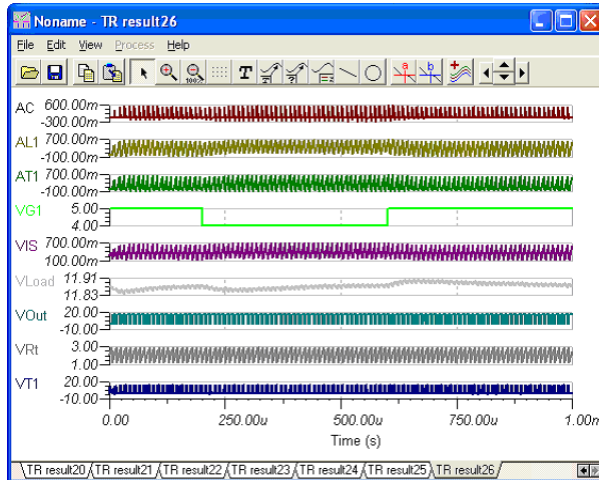
Load the example UC3842 Boost Converter (5\_12V Input Step.TSC file in the SMPS folder). The schematic design is the same as above. To see the input step waveform, double-click on the VG1 voltagegenerator on the left. The following dialog box will appear:





According to this, the input line voltage is 5V. This is converted by the SMPS circuit to 12V.

Now click on the Signal line of the above dialog and then the  button. The following signal in the Signal Editor will appear:



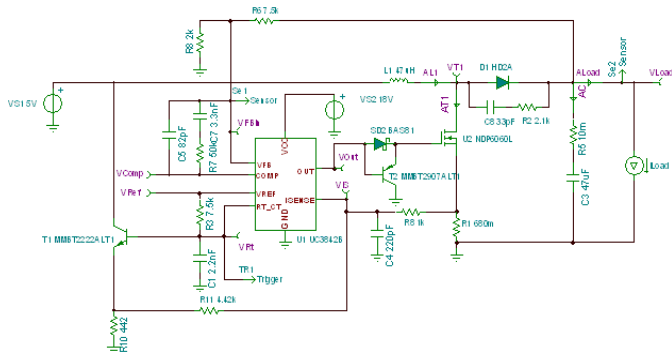
According to the waveforms, the input voltage will decrease from 5V to 4V for a  $T2=400\mu s$  time; and the starting edge (T1) and the ending edge (T3) of the pulse are 1us.

To see the response of the circuit, let's invoke and run the Transient analysis from the Analysis menu.

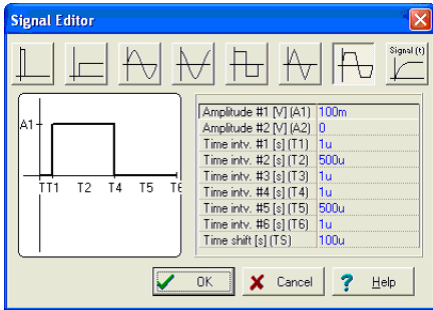
Load step analysis

Another standard analysis is to determine the SMPS response to a fast load change. Using simulation, the response to load changes is obtained by adding a current pulse to the load and analyzing the output and other voltages. Since the load change is relative to the steady state we can start it from the steady state initial values calculated by TINA's steady state solver.

Now load the example UC3842 Boost Converter. Find and load the file 5\_12V Load Step.TSC file from TINA's SMPS. The schematic design is the same as above, except for the ILoad Current generator on the output.

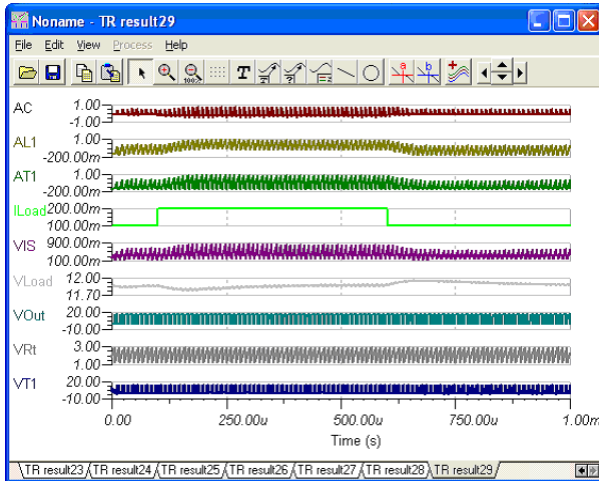


If you double-click on the ILoad generator and check its waveform, you will see that the DC part is 100mA while the pulse is



is 100mA in amplitude and 500us in width. Accordingly, the 100mA load current will rise to 200mA and then decrease to 100mA again. Note that the output current is shown through the ALoad current arrow at the output.

Let's run Transient from the Analysis menu and see the result.



#### 4.6.4 Stress Analysis

Stress Analysis can check parts for stress conditions such as maximum power dissipation and maximum voltage and current limits. You can set these parameters in the property window of the parts or in the catalog. This kind of analysis is also called Smoke analysis, because overloaded parts often emit smoke.

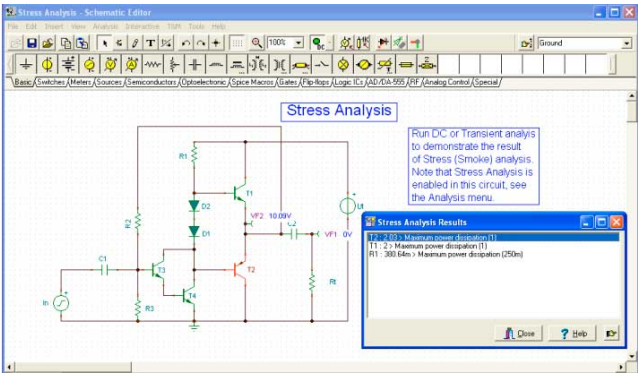
You can also run Stress Analysis by setting the Stress Analysis Enabled checkbox in the Analysis | Option dialog or on the Analysis menu.

When running DC or Transient Analysis from the Analysis menu, a list of components will appear, along with the parameters exceeding maximum limits.

If you click a component in the list, the corresponding component on the schematic diagram will be selected and turned red.

The maximum values of the components can be set in the component property dialogs or in the component catalog parameter dialogs. Both can be entered by double-clicking on the components. Before running an analysis, check and set the maximum values of the components in your circuit.

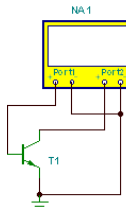
As an example of Stress Analysis, open the file Stress Analysis.TSC from TINA's EXAMPLES folder and run DC.Calculate Nodal Voltages and Transient Analysis from the Analysis menu or the corresponding interactive modes. In the following figure, you can see the result of Stress Analysis in DV interactive mode.



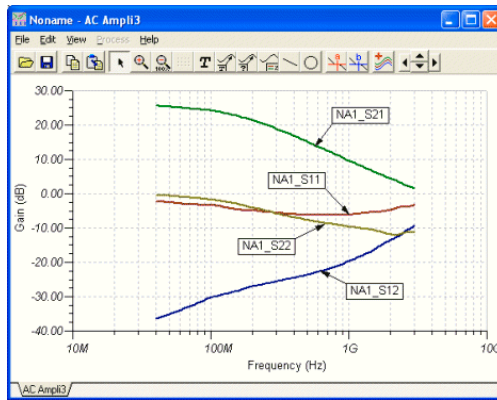
Apparently the power dissipation of T1, T2 and R1 exceed maximum limits allowed for these parts.


### 4.6.5 Network Analysis

TINA helps you perform network analysis and determine the two-port parameters of networks (S, Z, Y, H). This is especially useful if you work with RF circuits. Results can be displayed in Smith, Polar, or other diagrams. You can assign the two ports needed for Network Analysis with the Network Analyzer component to be found on the Meters component toolbar. As an example open the circuit EXAMPLES\RF\SPAR\_TR.TSC.



To analyze this circuit run Analysis/AC Analysis/Network Analysis. The amplitude diagram is as follows:

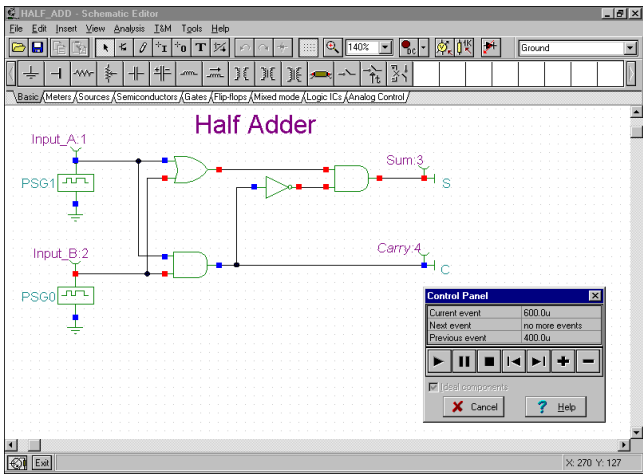


Note that we have added the labels to the curves using the  Auto label tool of the diagram window. For more details on the Network Analysis see the “Network Analysis and S-parameters” chapter of the advanced topics manual.

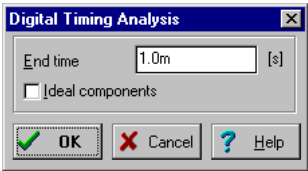
## 4.6.6 Analyzing a Digital Circuit with TINA’s Digital Engine

Let’s test a digital circuit. Open the file **HALF\_ADD.TSC** from the EXAMPLES folder. Start the *Analysis | Digital Step-by-Step* command. A control panel will appear and you can examine the behavior of the circuit step-by-step by pressing the Step Forward button. Press the Play button for free-running mode. At each node a small box will indicate the logic level (Red for High, Blue for Low, Green for High Z, Black for undefined) as the circuit is clocked.

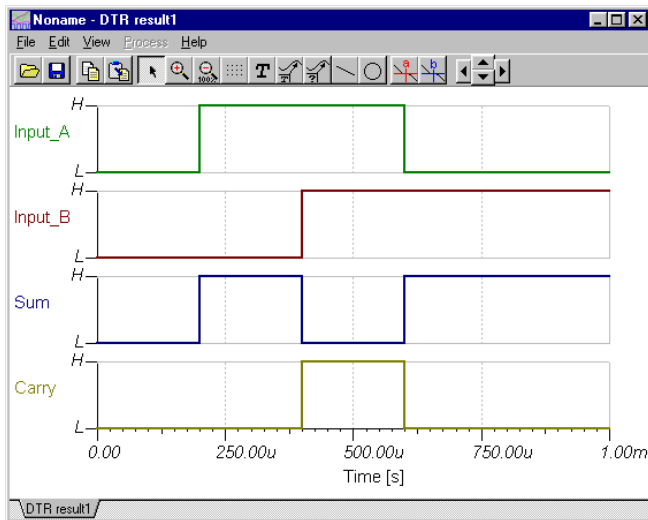
The picture below shows an intermediate state of the display.



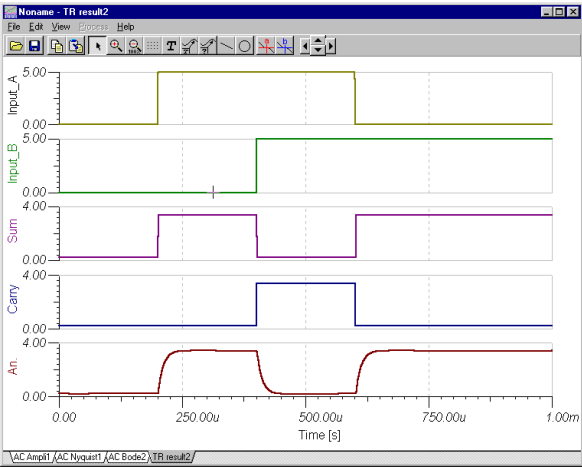
Now let's examine the transient behavior of the circuit. Selecting the *Analysis | Digital Timing Analysis* command, brings up this menu:



The result is shown on the timing diagram following.



You could also select *Transient...* instead of *Digital Timing Analysis*, in which case the program would carry out an analog analysis, giving the detailed continuous waveforms and voltages instead of idealized logic levels. Note that circuits which contain only digital components can be analyzed by both digital and analog methods. Circuits containing both analog and digital components, on the other hand, can only be analyzed by the analog method. Next, load the file **EXAMPLES\HALFADMX.TSC**. Because this circuit has two passive parts (a resistor and a capacitor), *TINA* must use analog (or mixed-mode) Transient analysis. The result is the following time response.



**NOTE:**

You can set the order of the curves by simply appending a colon (:) character and a number to the output name. This is particularly important when presenting the results of digital analysis, where each output is displayed as a separate diagram. For example, if you have outputs named OutA, OutB, Carry, and Sum, you can ensure that they will be displayed in the order given by using the labels OutA:1, OutB:2, Carry:3, and Sum:4.

The results of a purely analog analysis normally appear in one diagram: however, you can force TINA to display the results as separate diagrams, in the order you desire, by using the labeling method described above. You must use the View | Separate Curves command in the Diagrams window to separate the curves. If you don't use this labeling method, TINA presents the curves in alphabetical order.



## 4.6.7 Analyzing a Digital Circuit Using Digital VHDL Simulation

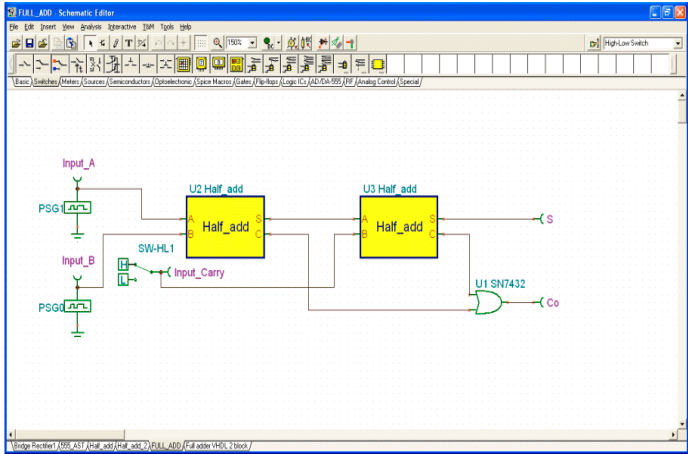
VHDL (Virtual Hardware Description Language) is an IEEE-standard hardware description language used by electronic designers to describe and simulate their chips and systems prior to fabrication

TINA v7 and higher versions now also include a powerful digital VHDL simulation engine. Any digital circuit in TINA can be automatically converted into VHDL code and analyzed as a VHDL design. In addition you can analyze a wide range of hardware available in VHDL and define your own digital components and hardware in VHDL. The great advantage of VHDL is not only that it is an IEEE standard hardware description language, but also that it can be realized automatically in programmable logic devices such as FPGAs and CPLDs.

TINA can generate synthesizable VHDL code along with the corresponding UCF file if the Generate synthesizable code checkbox is set in the Analysis/Options menu. You can save the created VHD and UCF files with the “Create VHD & UCF File” command in the T&M menu. You can read the files with the free Xilinx’s Webpack and generate the bit stream file describing the implementation of the design and then upload it to Xilinx FPGA chips. You can find the detailed description of this topic in the Advanced Topics user manual.

Before realizing a VHDL design, either with discreet components or FPGA, you need to verify it with simulation. This is very similar to the simulation with TINA’s Analysis | Digital Timing Analysis command using TINA’s built in digital models. To work with VHDL, you should just use the Digital VHDL Simulation command from the Analysis menu. Let’s examine some aspects of the VHDL simulation.

To do our first VHDL analysis, Open the FULL\_ADD.TSC circuit from the EXAMPLES/VHDL folder. The following circuit will appear:



This circuit is a combination of two VHDL half adder blocks (macros) and a discrete OR gate.

If you doubleclick on either of the HALF adder blocks and then press the Enter Macro button, the following window will appear:

```
TINA VHDL Editor

-- TINA VHDL Macro Description Begin
-- entity_name: e_Half_add_entity;
-- arch_name: a_Half_add_arch;
-- ports: A,B,S,C;
--
-- TINA VHDL Macro Description End

LIBRARY ieee, tina;
use ieee.std_logic_1164.all;
use std.textio.all;
USE tina.primitives.all;

-- entity section
--
ENTITY e_Half_add_entity IS PORT(
  A : IN std_logic;
  S : OUT std_logic;
  C : OUT std_logic;
  B : IN std_logic );
END e_Half_add_entity;

-- architecture section
--
ARCHITECTURE a_Half_add_arch of e_Half_add_entity IS

  signal N5 : std_logic;
  signal N6 : std_logic;

BEGIN

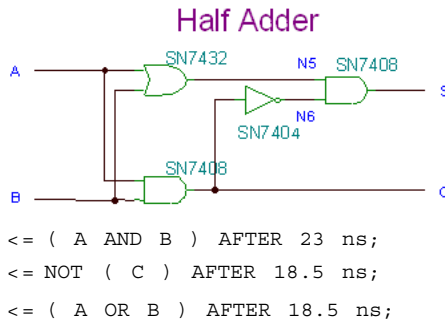
  S <= ( N5 AND N6 ) AFTER 23 ns;
  N6 <= NOT ( C ) AFTER 10.5 ns;
  C <= ( A AND B ) AFTER 23 ns;
  N5 <= ( A OR B ) AFTER 10.5 ns;

END a_Half_add_arch;
```

Note that the essential VHDL code of the half adder is at the bottom and it is only

```
S <= ( N5 AND N6 ) AFTER 23 ns;
N6 <= NOT ( C ) AFTER 18.5 ns;
C <= ( A AND B ) AFTER 23 ns;
N5 <= ( A OR B ) AFTER 18.5 ns;
```

At first glance, the code may look a bit strange, but it in fact is a machine translation of our half adder, assembled from gates in 4.6.1. Introducing the node names N5 and N6 as shown on the figure below, it is clear that



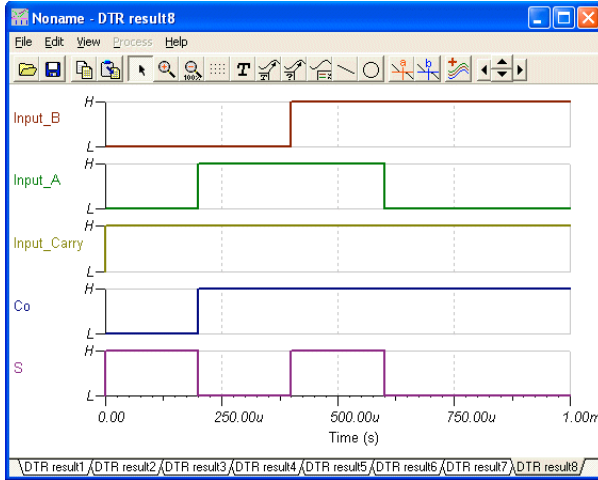
and therefore

```
S <= ( N5 AND N6 ) AFTER 23 ns;
```

You might find it odd that in the VHDL code in the box, S appears to be calculated from N5 and N6 even before N5 and N6 have been calculated. This is valid, however, because VHDL is a concurrent language and the order of the lines does not mean the order of execution. The delays are taken from the given discrete gates, but they will be replaced by the synthesizer program if the circuit is realized on an FPGA chip.

Now select Digital VHDL Simulation from the Analysis menu and press

OK. The following diagram will appear:



A great feature of TINA's VHDL is that you can not only view the VHDL code of each component, but you can edit and run them immediately. Let us replace the 4 line VHDL code -

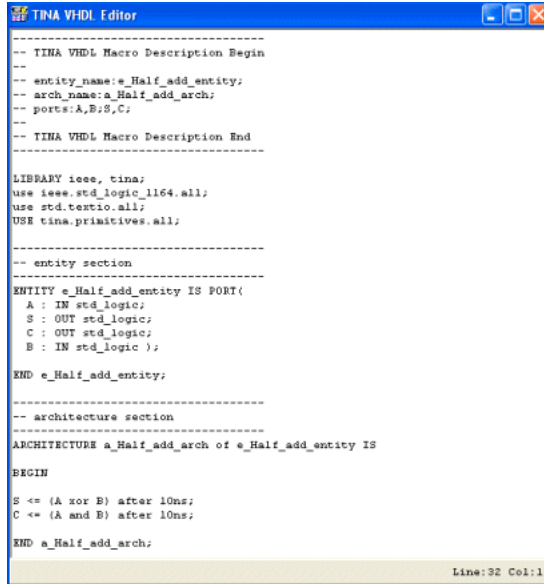
```
S <= ( N5 AND N6 ) AFTER 23 ns;
N6 <= NOT ( C ) AFTER 18.5 ns;
C <= ( A AND B ) AFTER 23 ns;
N5 <= ( A OR B ) AFTER 18.5 ns;
```

with this simpler 2 line code

```
S <= (A xor B) after 10ns;
C <= (A and B) after 10ns;
```

This is easier to understand. In fact, if one of the A or B inputs are true, the sum S is True, while if both are true, S is False. (A xor B), while in this case the Carry bit is True. (A and B).

After editing the content of the VHDL blocks, they should look like this:



```

TINA VHDL Editor

-- TINA VHDL Macro Description Begin
--
-- entity_name: e_Half_add_entity;
-- arch_name: a_Half_add_arch;
-- ports: A,B,S,C;
--
-- TINA VHDL Macro Description End
--
LIBRARY ieee, tina;
use ieee.std_logic_1164.all;
use std.textio.all;
USE tina.primitives.all;

-- entity section
=====
ENTITY e_Half_add_entity IS PORT(
  A : IN std_logic;
  S : OUT std_logic;
  C : OUT std_logic;
  B : IN std_logic );
END e_Half_add_entity;


-- architecture section
=====
ARCHITECTURE a_Half_add_arch of e_Half_add_entity IS
BEGIN

  S <= (A xor B) after 10ns;
  C <= (A and B) after 10ns;

END a_Half_add_arch;

Line:32 Col:1

```

Now close the edit window by pressing  on the Schematic Editor toolbar, select Digital VHDL Simulation from the Analysis menu, and press OK. The diagram that is drawn will be practically identical to the previous diagram.

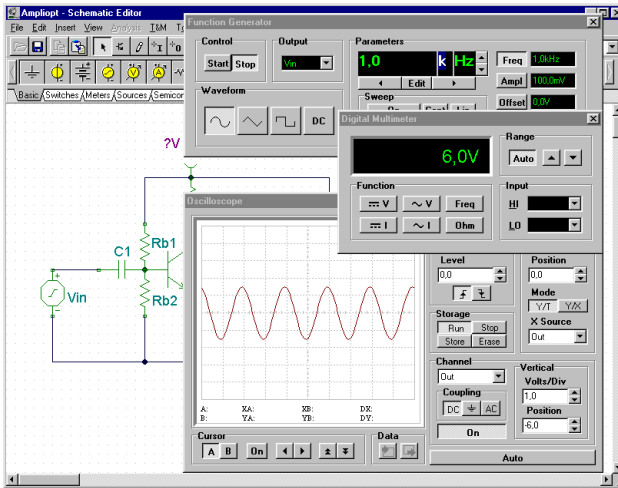
**NOTE:**

In TINA of course you can make your own VHDL macros. This is described in chapter 5 under **5.5 Adding VHDL macros to TINA**.

## 4.6.8 Testing your Circuit with Virtual and Real Time Instruments

*TINA* lets you test and tune your circuit not only with the generators and analysis windows that you have used so far, but also with virtual and real-time measurements. Using the T&M menu you can place realistic virtual instruments on screen which will automatically replace

the generator and analysis windows. You can control the settings of these instruments and immediately see the results, just as you would in a real lab. *TINA* normally simulates measurements with its analysis engine, but if you have *TINA*'s supplementary hardware, you can simply switch to the Real Measurement mode (using Option of the T&M menu). Now you can work with the same on-screen instruments and settings and you will be making real measurements on a real circuit.



To start with virtual measurements, load the circuit **AMPLIOPT.TSC** from the examples directory. Select the T&M menu and place a Multimeter, Function Generator, and an Oscilloscope on screen. Press Run in the Oscilloscope panel. A distorted sine curve will appear on the oscilloscope screen. Click on the multimeter = button. The multimeter will show only about 0.7 V volts at the collector (Out) - this is the reason for the distortion. Now double-click on the  $Rb1$  resistor. The dialog box of resistor properties will appear. Click on the Resistance field and then change the value with the arrows on the right hand side of the dialog box until the multimeter shows about 6V. You can change the step size of the buttons by entering it into the edit box under the *down* arrow. You can also define a hotkey to the up and down arrow by selecting it from the upper and lower listboxes. Note that while the interactive mode is On, and once a hotkey is defined, the value of the resistor can be changed directly by pressing the appropriate key without opening the property dialog. You can assign hotkeys to

most component values in TINA, including switches. To avoid accidental changes, the hotkeys for component values will only work while TINA is in the interactive mode. Switch positions, however, can be changed before activating the interactive mode in order to set their initial position. Once the collector voltage reaches 6V, close the property editor dialog box (if it is still open) and press Run on the oscilloscope. Set the vertical position to -6 V and use the horizontal and vertical settings to scale the curve for best appearance. The distortion will no longer be visible.




Press the *Ampl.*-button on the generator. The last amplitude value will appear in the large numeric display field of the generator. Use the vertical arrows beside the display to change the amplitude. You will see the sine wave become distorted again as you increase the amplitude, with the maximum input at about 500mV. Now change the waveform from sinusoidal to triangle and then to square wave. Vary the frequency of the function generator to explore the frequency domain over which the circuit performance is acceptable.

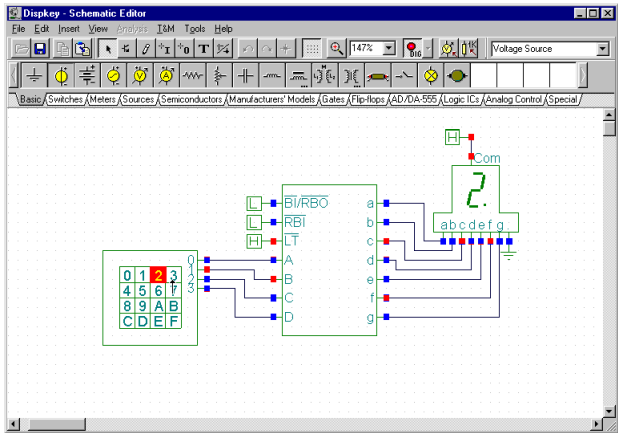
**NOTE:**

The virtual instruments under the T&M menu are not to be confused with the virtual instrument *components* on the Meters component toolbar. Some of the virtual instrument components are used in the interactive mode of the program, discussed in the next section. They are also used to assign outputs for the various analysis modes under the Analysis menu. The Oscilloscope and Signal Analyzer virtual instrument components have a small screen and their purpose main function is to be used with our 3D circuit analyzer program, EDISON.


## 4.6.9 Testing your Circuit in Interactive mode

When everything is in order, the ultimate test of your circuit is to try it in a “real life” situation using its interactive controls (such as keypads and switches) and watching its displays or other indicators. You can carry out such a test using *TINA*’s interactive mode. Not only can you play with the controls, but you can also change component values and even add or delete components while the analysis is in progress. The

interactive mode is also very useful for educational and demonstration purposes, for tuning circuits interactively and for interactive circuits which you cannot test otherwise, e.g., circuits with switches, relays, or microcontrollers. First select the interactive mode required (DC, AC, TR, DIG or VHDL) with the  button, then press the  button. XX can be DC, AC, TR, VHD etc. depending on the mode, set by the  button. You can also select the required interactive mode with the DC, AC, Transient, ...VHDL commands of the Interactive menu. You can start the interactive simulation with the Start command of the Interactive menu and stop it with the Stop command (The Start command will turn into Stop when the interactive simulation is started). Now the displays and indicators in your schematic will reflect whatever you do with the controls. In addition to displays, *TINA* has special multimedia components (light bulb, motor, LED, switch, etc.) which respond with light, motion and sound. Let's see a few examples.



#### 4.6.9.1 Digital Circuit with a Keypad

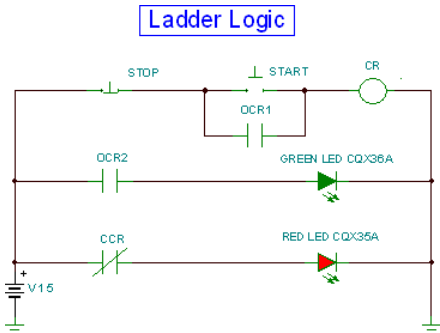
To try out the interactive mode, load the **DISPKEY.TSC** circuit from the *EXAMPLES\MULTIMED* folder. The circuit is shown below. Select the *Digital* mode using the  button, and then press the DIG button (the button will turn light green).



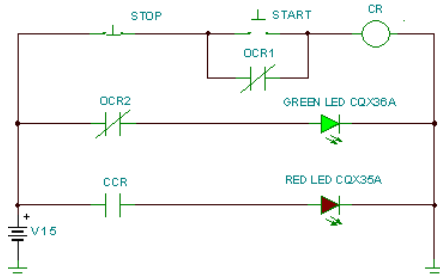


Initially the red LED will light. If you click on the START button (click when the cursor changes into a vertical arrow), OCR1 will close and stay closed (since the current flowing through OCR1 will keep magnetizing the relay coil CR). Now the green LED will light, OCR2 will open, and the red LED will turn off. If you now click on the STOP button, you will break the self holding circuit and the relay CR will release, the red LED will light again, and the green LED will turn off.

You can make it easier to operate switches if you “assign” them to “hotkeys” on the keyboard (your PC’s keyboard). Double-click on a switch when the cursor has turned into a hand symbol. To assign a hotkey, select a letter or number on the list at the Hotkey field of the property dialog of the switch.



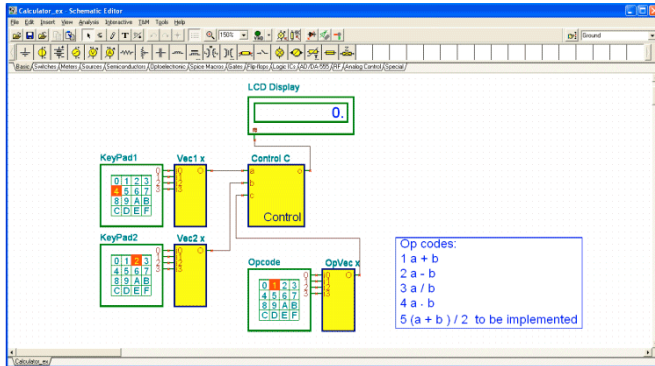
Ladder logic: Initial state or after clicking the STOP button.




State after clicking the START button

#### 4.6.9.4 VHDL Circuits

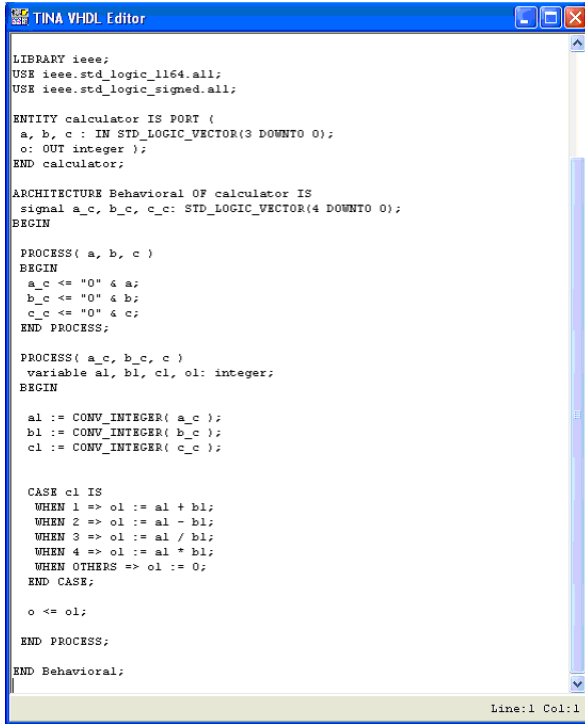
A great feature of TINA is that you can not only test but also modify VHDL circuits on the fly including the VHDL code itself. Let's see this with the example Calculator\_ex.TSC in TINA's Examples/VHDL/Interactive folder.



This is a special calculator circuit controlled by the Opcode keypad. For the Operation codes 1, 2, 3 and 4, it realizes a basic four function calculator, complete with +, -, /, and \* basic arithmetic operations. Further operations can be added through modifying the VHDL code inside the Control unit. First press the  button; as the Opcode is 1, you should see 4+2=6 on the LCD display. Try the other Opcodes with different settings on KeyPad1 and KeyPad2.

Now let's implement the operation to be assigned to Opcode 5.

Double-click on the Control box and press Enter Macro. The VHDL code of the component will appear.



```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_signed.all;

ENTITY calculator IS PORT (
  a, b, c : IN STD_LOGIC_VECTOR(3 DOWNTO 0);
  o: OUT integer );
END calculator;

ARCHITECTURE Behavioral OF calculator IS
  signal a_c, b_c, c_c: STD_LOGIC_VECTOR(4 DOWNTO 0);
BEGIN

  PROCESS( a, b, c )
  BEGIN
    a_c <= "0" & a;
    b_c <= "0" & b;
    c_c <= "0" & c;
  END PROCESS;

  PROCESS( a_c, b_c, c )
  variable a1, b1, c1, o1: integer;
  BEGIN
    a1 := CONV_INTEGER( a_c );
    b1 := CONV_INTEGER( b_c );
    c1 := CONV_INTEGER( c_c );

    CASE c1 IS
      WHEN 1 => o1 := a1 + b1;
      WHEN 2 => o1 := a1 - b1;
      WHEN 3 => o1 := a1 / b1;
      WHEN 4 => o1 := a1 * b1;
      WHEN OTHERS => o1 := 0;
    END CASE;

    o <= o1;


  END PROCESS;
END Behavioral;
  
```

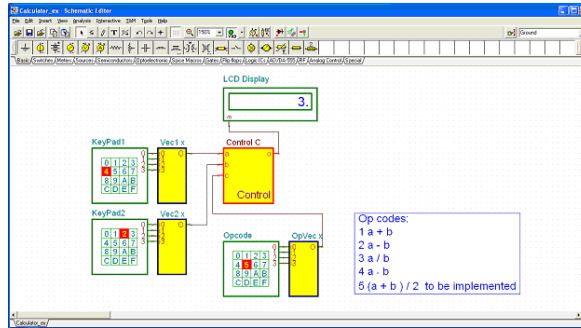
Line:1 Col:1

The actual calculations are made in the CASE statement at the end of the VHDL code. Let's modify the code like this:

```

CASE c1 IS
  WHEN 1 => o1 := a1 + b1;
  WHEN 2 => o1 := a1 - b1;
  WHEN 3 => o1 := a1 / b1;
  WHEN 4 => o1 := a1 * b1;
  WHEN 5 => o1 := (a1 + b1)/2;
  WHEN OTHERS => o1 := 0;
END CASE;
  
```

Close the VHDL Editor Window and press the  button. Set 5 at the Opcode keypad and you should see the average of KeyPad1 and KeyPad2 settings on the LCD Display.



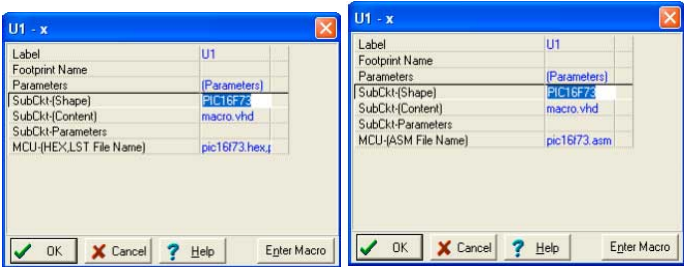
#### 4.6.9.5 Microcontroller (MCU) Circuit


To test circuits with programmable devices requires special development software that permits a high degree of interactivity. This calls for debugging software that can test the code running in the device step-by-step. TINA v7 and higher versions support PIC microcontrollers; support for other MCUs are constantly being added to TINA. Even though you can add your own microcontrollers described by VHDL, the built in microcontrollers in TINA are precompiled for higher performance. Their VHDL code is not visible to the user.

In any event, you can see, modify, and debug the program running in any of the supported processors, and, of course, you can make and run your own code.

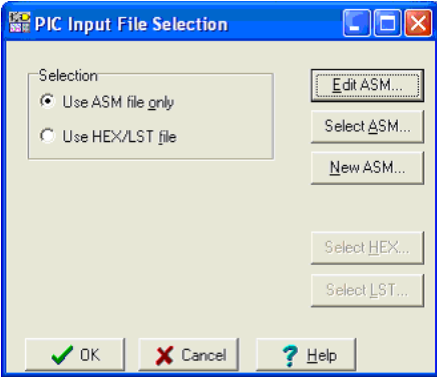
There are two ways of providing the program for microcontrollers in TINA. You can use the binary code and debug file made by any standard compiler (e.g., MPLAB for PICs), or you can just load your assembly code to run and debug directly in TINA using its built in assembler-debugger.

To load the code into the MCU, double-click on the schematic symbol. One of the following dialogs will appear:



Click on the last MCU File name line and press the  to proceed.

The following dialog will appear:

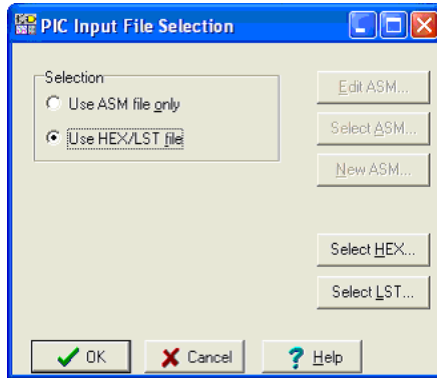


Here you can see and edit the ASM code in the MCU, select another ASM code file, or create a New ASM directly in the editor that will appear when you press the New ASM button.

If, however, you switch to the Use HEX/Lst file option, you can select the binary (HEX) file you want to run and the LST file to be used for debugging, as shown in the dialog below.

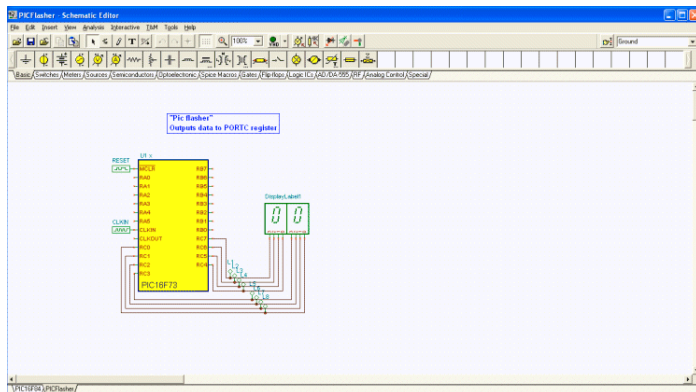
**NOTE:**


The HEX and LST files should be generated by an appropriate compiler (normally provided free by the MCU manufacturer. However, TINA has a built in compiler for all supported MCUs, so you can directly use your ASM source code.

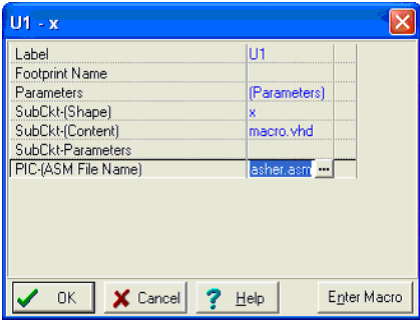



#### 4.6.9.6 Example PIC Flasher

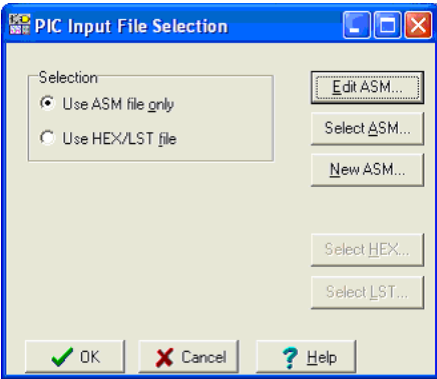
Now let us run a microcontroller application and see how to modify its code. Load the PIC Flasher.TSC circuit form the Examples\VHDL\PIC\Asm folder. The following schematic will appear with the 16F73 PIC microcontroller.




This circuit is simply counting forward one count at a time. Press the  button to see how it works. The display should step forward one-by-one.

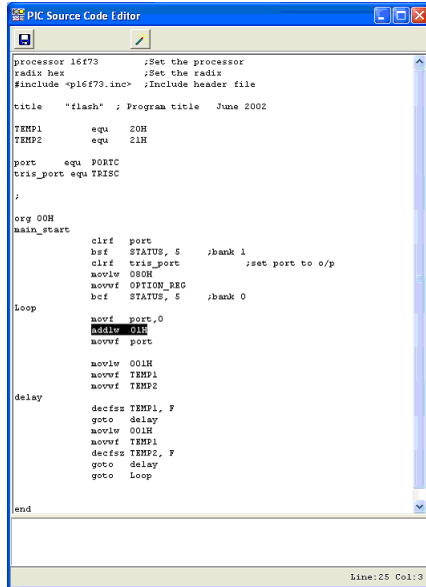


Now let's release the  button and modify the code to count by 2.



Double-click the MCU, and click on the  button in the dialog below -





```

PIC Source Code Editor

processor 16f73      ;Set the processor
radix hex           ;Set the radix
#include "pic16f73.inc" ;Include header file

title "flash" ; Program title   June 2002

TEMP1 equ 20H
TEMP2 equ 21H

port equ PORTC
tris_port equ TRISC

;

org 00H
main_start
    clrf port           ;bank 1
    bcf STATUS, 5
    clrf tris_port      ;set port to o/p
    movlw 000H
    movwf OPTION_REG
    bcf STATUS, 5      ;bank 0

loop
    movf port,0
    addlw 01H
    movwf port

    movlw 001H
    movwf TEMP1
    movwf TEMP2

delay
    decfsz TEMP1, F
    goto delay
    movlw 001H
    movwf TEMP1
    decfsz TEMP2, F
    goto delay
    goto Loop

end

Line:25 Col:3

```



Press the Edit ASM button. The ASM code of the MCU will appear in the MCU Source code editor.

Now lets make the following change in the code. Change the instruction (selected above) in line 25 (you can see the line number in the right bottom corner of the code editor window) from

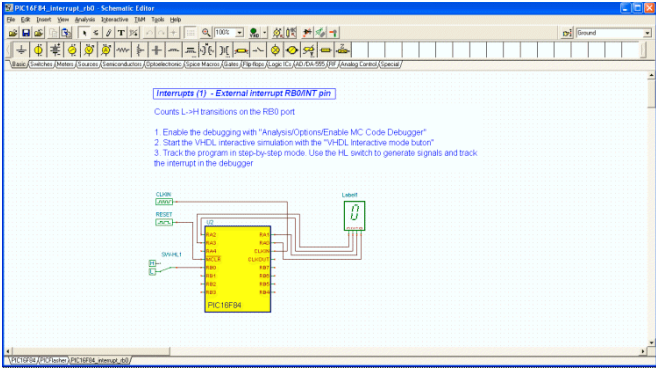
```
addlw    01H
```

to

```
addlw    02H
```


Save the changed code by pressing the  icon and close the open MCU windows. If you press the  button, now the increment will be 2!

Note that the changed code will be automatically saved in the TINA.TSC file.



#### 4.6.9.7 Example PIC Interrupt handling

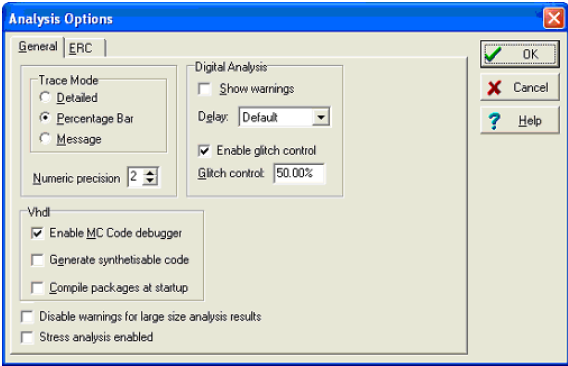
Now let's see another application with some more interactivity. Load the PIC16F84interrupt\_rbl0.TSC example from the Examples\VHDL\PIC\Asm folder.

Press the  button. At first glance, it appears that nothing is happening.

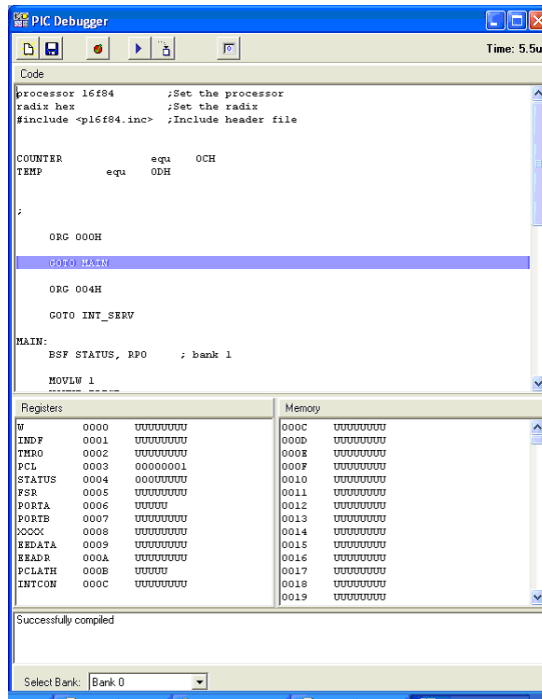
However, if you click on the SW-HL1 switch, the display will step forward by 1 each time that the switch changes from Low to High. This is realized with the interrupt handling capability of the PIC16F84.

Now let's see the operation in more detail using the interactive ASM debugger.

To activate the debugger, select Option on the Analysis menu. Set the "Enable MCU Code debugger checkbox," as shown below in the Analysis Options dialog box.



The MCU debugger will appear if you press the button:



Here is a short description of the MCU debugger dialog.

On the top line there are the following controlling icons:



**New.** Clear the debugger. You can enter, automatically compile, debug and run new code. This icon appears only when the MCU code is given in ASM source format. (Use ASM file only option in the “MCU input file selection” dialog available from the “MCU code” line MCU property dialog)



**Save** the actual code into the TINA TSC file. Note that you can edit the code in the Debugger. After editing, it is automatically recompiled (after requesting confirmation). This icon appears only when the MCU code is given in ASM source format.



**Save the ASM file** (source code) of the MCU code. This icon appears only when the MCU code is given in ASM source format.



**Save the executable binary HEX file** of the MCU code.

## Using the MCU Debugger



Save the LST debug file of the MCU code.



Toggle breakpoint: Inserts or removes breakpoints in the selected line. Click on the line where you want to place or remove the breakpoint before clicking on the icon.



Run the code in the debugger continuously. The lines being executed will be highlighted and the code is scrolled to the being executed will be highlighted and the code is scrolled to the line being executed.




Trace into. Step by step execution. Each time you press this button one command of the program is executed.



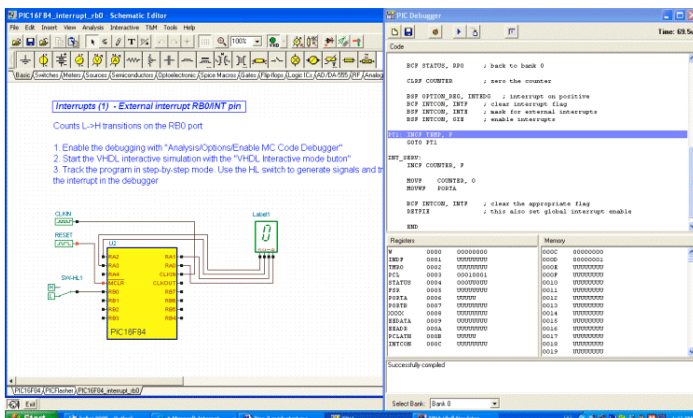
Stop Halts program execution.

The Code window (below the control icons) displays the ASM code. The next actual command is highlighted with blue.

The actual content of the registers and memory locations of the MCU are shown in the lower part of the screen.


Let's follow the program execution step-by-step by pressing the  trace into button. After around 14 clicks, we get to the PT1: label, where the program seems to be in an infinite loop.

```
PT1:  INCF  TEMP,  F
      GOTO  PT1
```



The screenshot shows the PIC16F84 Interrupt.asm schematic editor on the left and the PIC debugger on the right. The schematic editor displays a circuit diagram with a PIC16F84 microcontroller connected to a pushbutton (SW14.1) and a LED (LED1). The PIC16F84 is configured with various pins and internal components. The PIC debugger window on the right shows the assembly code for the Interrupt.asm file. The code includes initialization for the STATUS register, clearing the counter, and setting up the interrupt. The current instruction being executed is the PT1 label, which increments the TEMP register and jumps back to PT1, forming an infinite loop. The debugger also displays the registers and memory locations of the MCU.

Now click on the SW-HL1 switch and change it to High. (You should click when the cursor changes into an upward pointing arrow ↑).

Return to the Debugger and click the  Trace into button twice. The program will recognize the interrupt and jump into the INT\_SERV: label.

```
INT_SERV:
    INCF    COUNTER,  F
    MOVF    COUNTER,  0
    MOVWF   PORTA
```

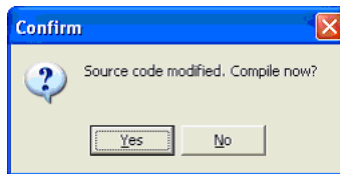
increment the COUNTER, and copy it to PORT A. The output will now be 1. After this, the program will return to the “infinite loop” at PT1.


## 4.6.9.8 Editing the Code in the Debugger


Now let's see how to make a small change in the program using the debugger. Duplicate the INCF COUNTER, F statement using Copy and Paste like so:

```
INT_SERV:
    INCF    COUNTER,  F
    INCF    COUNTER,  F
    MOVF    COUNTER,  0
    MOVWF   PORTA
```

No if you press the  the program will ask:





Press Yes and then press the  button again. Now the increment will be 2 at each Low-High change of the switch.

You can also check the circuit in the Debugger's continuous Running mode by pressing the  button. Even though the debugger will

run fast, you can still see the “infinite cycle” and the jump to the Interrupt server routine (INT\_SERV:) when you change the switch.

### 4.6.9.9 Making a Breakpoint



It is often essentially impossible to get to a certain place in the program since you’d have to single step a thousand times (if the program ever steps there in the first place). To get the program to run to a particular statement and halt there, you can tag the statement as a so called breakpoint. Now run the program in the Debugger’s continuous mode using the  Run command and the program will stop at the marked space before execution of the marked command.

To demonstrate this, click on the increment statement in our interrupt Service routine after the INT\_SERV: label and press the  Toggle break button.

Now press the  Run button. The program starts to run and falls into the “infinite loop.”

“infinite cycle” Even though you have set a breakpoint, the code will not stop since it does not pass the breakpoint. However, when you change the switch from Low to High the program will stop at the

```
INT_SERV:
    INCF COUNTER, F
```

statement. Now you can resume execution either step by step  *or* *with the*  Run command again.

You can find further multimedia examples in the EXAMPLES\MULTIMED directory.

## 4.7 Creating a Printed Circuit Board (PCB)

Once you have completed your circuit diagram, you can design a printed circuit board to manufacture your design. This is easy to do in TINA 7 and later versions, since PCB design is an integral part of the program.

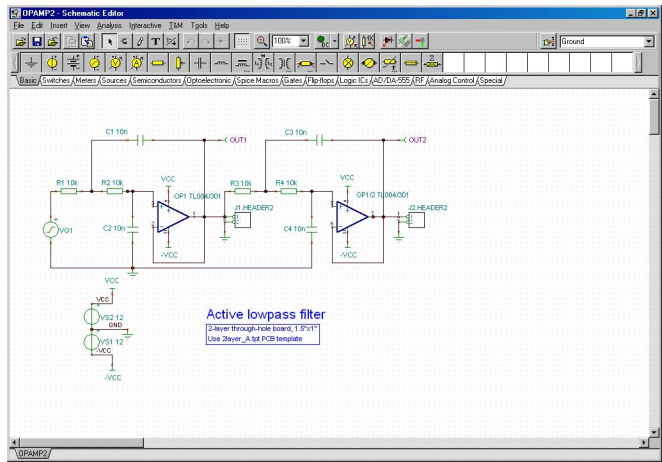
We'll learn about the PCB design process through a few examples.

The files from the different phases of the design examples have been saved in TINA's Examples/PCB directory using the following naming conventions:

* origin.tsc	original schematic file
*.tsc	schematic file backannotated ( after pin/gate swapping and renumbering )
* placed.tpc	design parameters set, components placed pcb file
* routed.tpc	net properties set and routed pcb file
* finished.tpc	optionally pin/gate swapped and renumbered, routed, silkscreen adjusted, documentation layers finalized pcb file

### 4.7.1 Setting and checking footprint names

To see the first example, open the opamp2.tsc project from TINA's Examples/PCB folder. The following schematic will appear:



The most important thing in PCB design is that every part in your schematic must have a physical representation with exact physical size. This is accomplished through so called footprints—drawings showing the outline and the pins of the parts.

TINA's footprint naming uses as a starting point the IPC-SM-782A (Surface Mount Design and Land Pattern Standard) and the JEDEC standard JESD30C (Descriptive Designation System for Semiconductor-Device Packages). (See <http://www.jedec.org/download/search/jesd30c.pdf>).

In TINA, we have already assigned default footprint names to all parts which represent real components.

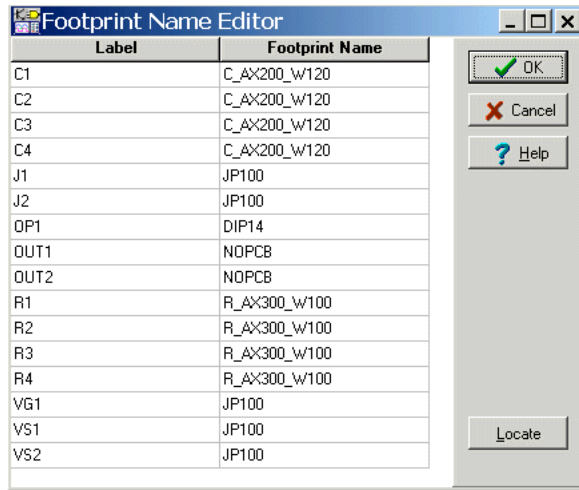
**NOTE:**

Some parts used for theoretical investigations ( for example, controlled sources) do not represent real physical parts so you cannot place them on a PCB. If your design contains such components, you should replace them with real physical parts.

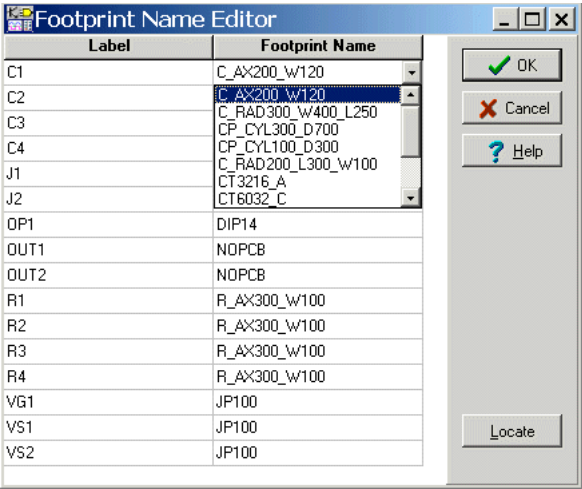


Of course there is no guarantee that the default physical representatives of the parts are the same as those needed by your design. There are two ways to check this.

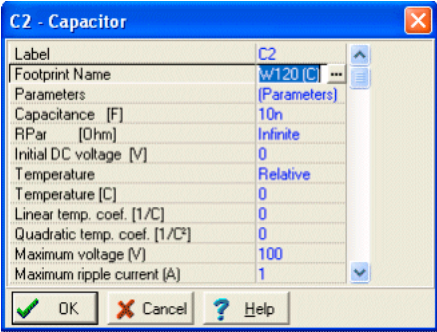
- 1) You can use TINA's "Footprint name editor" which you can invoke from the Tools menu. In this dialog you see all of TINA's components and the corresponding footprint names.




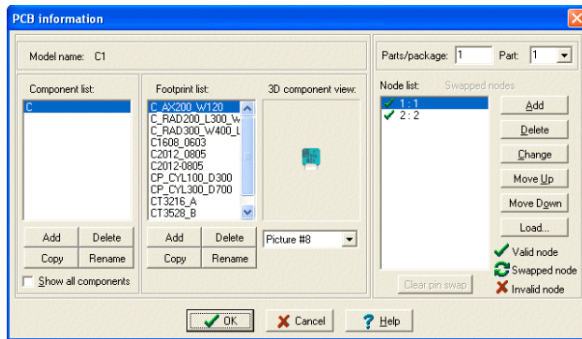
Clicking on the footprint name fields, you can select from the available footprint names. In the dialog, components that do not already have a footprint name association will be denoted by red characters and also by "???" in the footprint name field.



- 2) Alternatively you can double-click on each part and check the Footprint Name of the component property dialog.




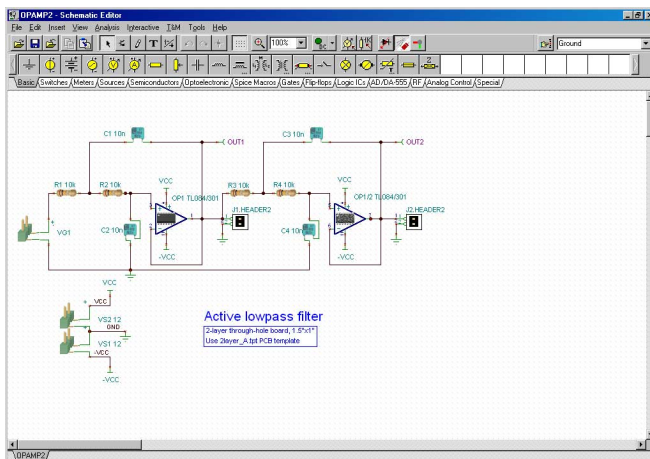
You can also click the  button in the Footprint Name line and see the “PCB information” dialog where you can select from the available footprint names. You can also see the 3D view of the different parts via the 3D package view field of the dialog.




If you find the footprint name you want on the list, click on it and press OK: you will be returned to the component property dialog with the selected footprint name in the Footprint name line. To confirm the change, press OK on the component property dialog again.

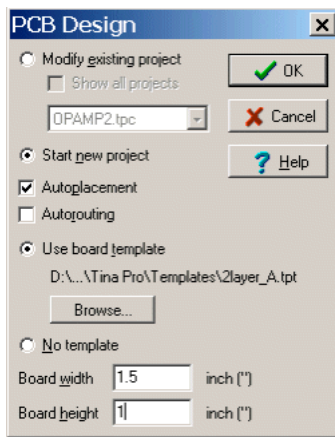
If you do not find the footprint name you want, you can add a new footprint using the Add buttons of the “PCB information” dialog. Press the Help button for more information.

When everything looks good, you can make a final check by clicking the  2D/3D view button or by simply pressing the F6 key. The 3D view of those components for which a physical representation has already been added will appear.



## 4.7.2 Invoking TINA PCB

Once each component has a satisfactory physical part association, we can proceed to PCB layout design. To do this, press the  button on TINA's toolbar (the last icon on the right) or select the "PCB Design" command on the Tools menu. Set the items in the PCB Design dialog as shown below.



Select "Start New Project", "Autoplacement" and "Use board template". With the Browse button find and select the 2layer\_A.tpt template files from TINA's Template. The settings are appropriate for a double-sided PCB.

If you use a template, you should set the level of manufacturing complexity. The following three levels of manufacturing technology are defined by the IPC-2221 generic standard.

- Level A : General Design Complexity**
- Level B : Moderate Design Complexity**
- Level C : High Design Complexity**

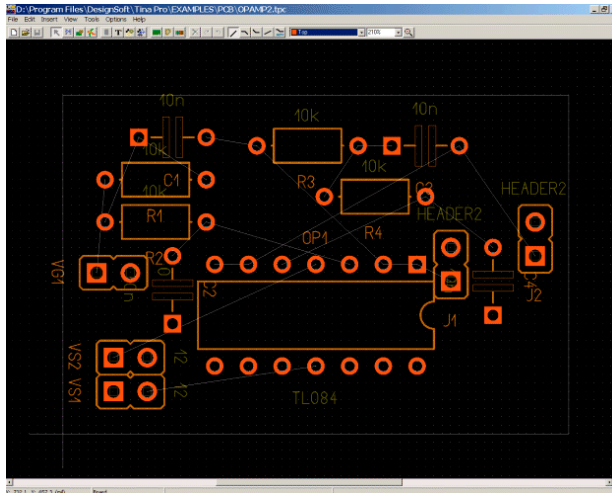
The template file specifies the number of layers and their properties: system grid size, autorouter settings, spacing and track width. The following templates are included with PCB Designer:

	Level	Routing Layers	Plane Layers	Routing	Spacing	
1layer_A.tpt	A	1	-	25	12 1/2	Allows one track between standard
2layer_A.tpt	A	2	-	25	12 1/2	DIP IC pin
2layer_B.tpt	B	2	-	8 1/3	8 1/3	Use for SMT or mixed-technology board
2layer_B_mm.tpt	B	2	-	0.1	0.2	
4layer_C_mm.tpt	C	2	2	0.1	0.15	For moderate and high density SMT boards

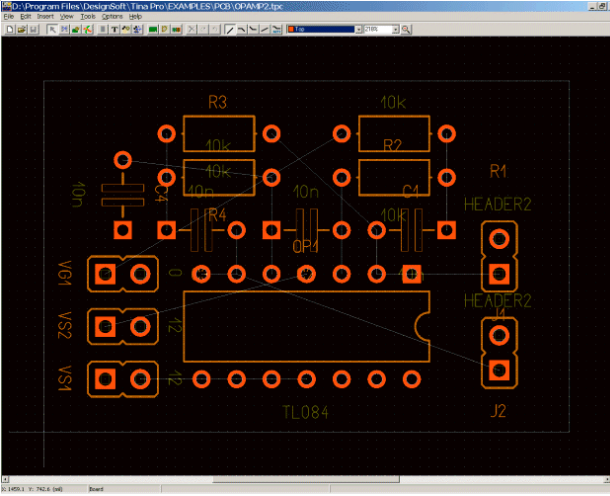
You can choose PCB template based on technology, density and package pitch.

Finally, you can set the size of the PCB board in inches or mm depending on the measurement unit settings in the View/Options dialog of TINA.

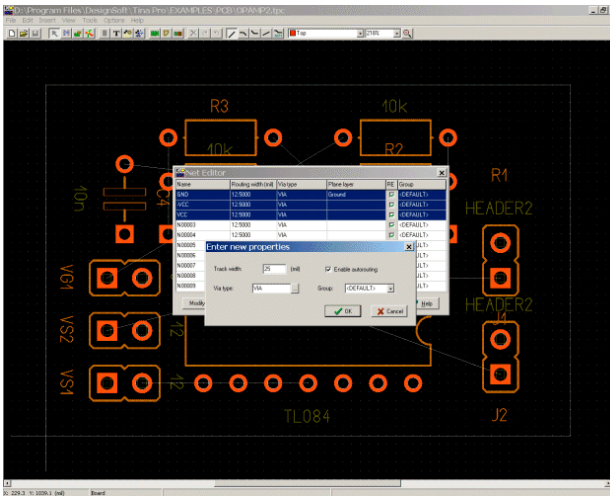
When everything is set properly, press the OK button and the PCB layout design will appear with all the components automatically placed on the PCB board.



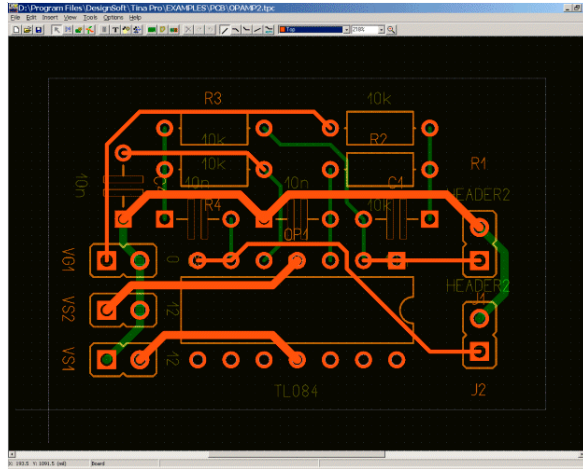
Now click and drag the parts to new positions as, shown on the figure below. (Find “opamp2 placed.tpc” to check your results.)



Press F4 to invoke the Net Editor and set nets routing width. First, click on “Modify all” and enter 12.5 into the “Track width” field. Then select power nets (Ground, VCC, -VCC) and set their widths to 25mil.



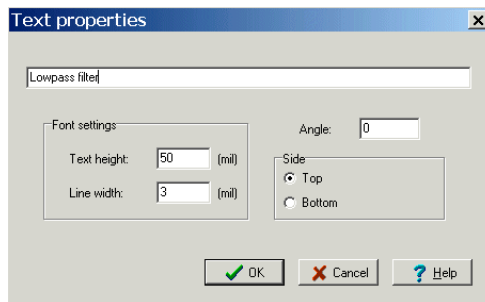
To automatically route the netlist, press the F5 button or select “Autoroute board” command from the Tools menu. The following screen will appear:



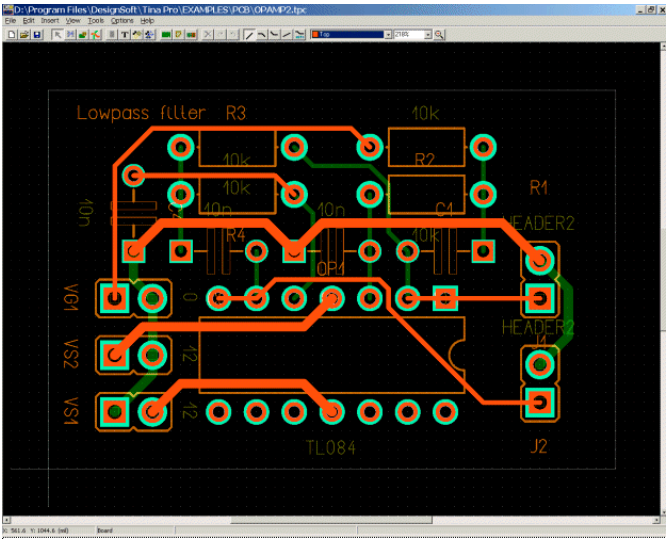
To see if everything is routed correctly, press F7 or select DRC (Design Rule Check) from the Tools menu. The following message will appear:



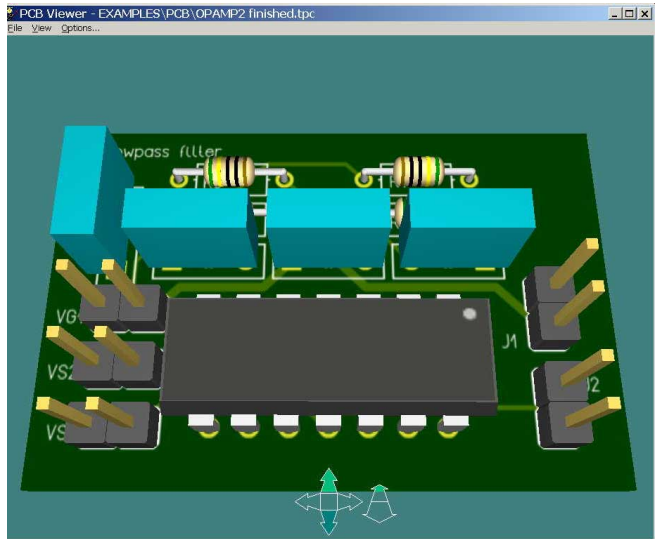
To finish our first simple design, let's add a text box to the silkscreen/assembly layer. To do this, click the T button on the toolbar. The following message will appear:



Enter the text into the empty upper field and press the OK button. The text will be attached to the cursor. Move it to the place shown on the picture below and press the left mouse button.



Finally, you can check your design in full 3D. To do this, press F3 or select 3D View from the View menu. After some calculation the following window will appear.





You can rotate the 3D model and zoom in and out with the control arrows at the bottom. You can display or hide these arrows through the Options menu with “*Use control arrows*” checkbox.

You can also rotate this 3D model by clicking with the mouse at any point, holding down the left button and moving the mouse. You can also move the camera forward or backward to see the whole design or just a part of it in more detail. To move the camera, hold down the right mouse button and move the mouse.

After this you can either print your design or create a Gerber file for a manufacturer.

To print use Print... from File menu.

To obtain Gerber files to direct a photoplotter, choose Export Gerber file from File menu. (Gerber option can be changed through Gerber output setting under Options menu.)

This example concludes the introduction to the use of TINA's PCB layout module. For more detailed information, a description of editing functions, the creation of multilayer PCBs, and more, see the TINA PCB Designer Manual.

We also suggest that you study the examples in TINA's EXAMPLES/PCB folder.



# USING SHEMATIC SUBCIRCUITS, SPICE AND VHDL MACROS AND S-PARAMETER COMPONENTS

In *TINA*, you can simplify a schematic by turning portions of it into a subcircuit. In addition, you can create new *TINA* components from any Spice subcircuit hardware described by VHDL or TouchStone format S-parameter file, whether created by yourself, downloaded from the Internet, or obtained from a manufacturer's CD. In this chapter, we show through text and examples how easy it is to do this in *TINA*.

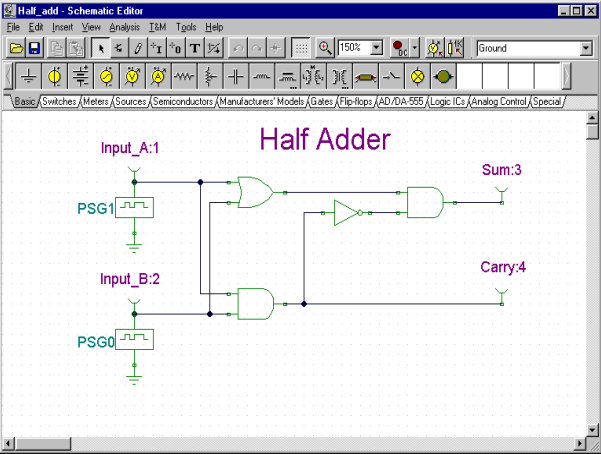
## 5.1 Making a Macro from a schematic

---

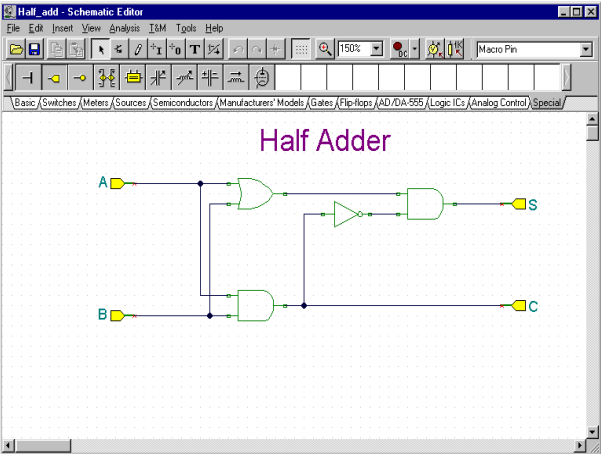
Using *TINA*'s macro facility, you can simplify schematics and hide clutter by turning portions of the schematic into a subcircuit. *TINA* automatically represents these subcircuits as a rectangular block on your schematic, but you can create any shape you like with *TINA*'s *Schematic Symbol Editor*.


You can convert any schematic diagram into a subcircuit - called a Macro in *TINA* - simply by adding the terminals and saving the new circuit in the special (\*.tsm) format.

Now let's see how to create a macro in *TINA* through an example. Load the Half Adder example (*Half\_add.tsi*) from the Examples folder of *TINA* and convert it into a macro.



Delete the old terminals and replace them with subcircuit terminals, called Macro Pins in *TINA*. You can find and select the Macro Pins under the Special component toolbar.

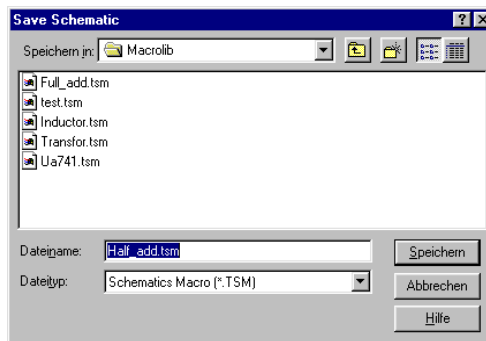


When you place Macro Pins, labels (such as Pin1, Pin2 etc.) are pre-filled in. Double click the Macro Pin and type in the new name in the label field. You can also drag the component with the mouse, or rotate it with the  $[+]$  and  $[-]$  keys or the  buttons.

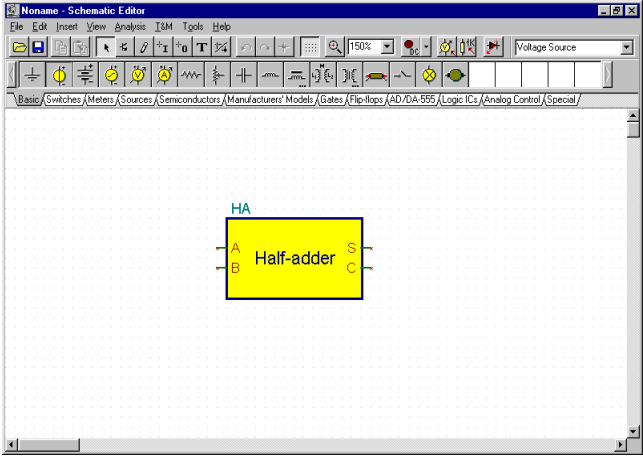
Next, create and save the new macro. Select the *New Macro Wizard* from the Tools menu. Set the Name to Half Adder (this will be displayed in the macro box that opens automatically), and set the Label to HA. This label will be displayed as the component label above the component. Note that you can leave this field blank if you don't want a component label.



When done, press OK. A Save dialog box will appear. Set Half Adder as the File name and press Save. Note that there is already a macro with a similar name (Half\_add.tsm). This has the same content as the one we just created, and is included for reference. You can also use it in the next section.



Now let's see how to insert a macro into a schematic and use it. Clear the circuit with **File | New** or by restarting *TINA*. Select **Insert | Macro**, then our newly created *Half adder.tsm*, and click Open.



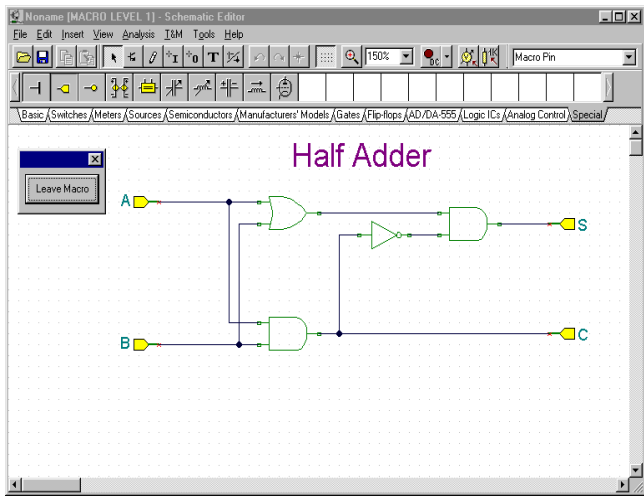
Our new macro will appear as a cursor. Move it to the center of the screen and click the left mouse button. The full symbol of the new macro will appear. Note that a rectangular schematic symbol has been automatically created, the macro name we specified is inside the rectangle, and the label name is above it.

Now you can add more components to the circuit, connecting them to the newly created macro, and start analysis as with any other circuit.

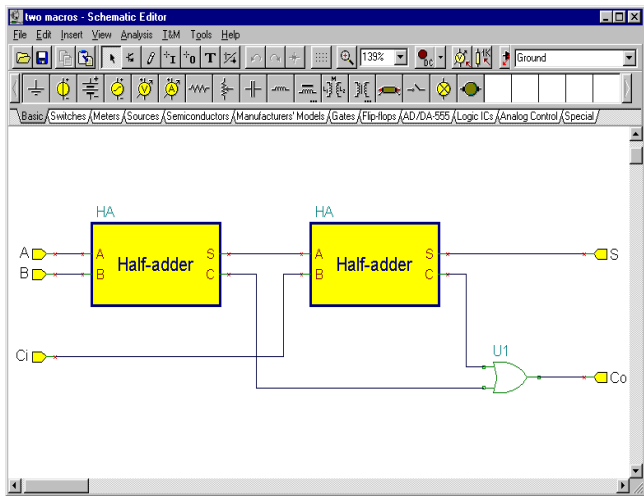
To check the content of the macro, double click on the symbol and *TINA* will display the model.

To return to the main circuit, click the Leave Macro button on the screen at the upper left, use the File/Leave Macro command, or the right click popup menu.


*TINA* allows a hierarchical macro structure; that is, macros can contain other macros inside, and so on. Let us use our half adder macro to create a full adder macro containing two half adder macros.

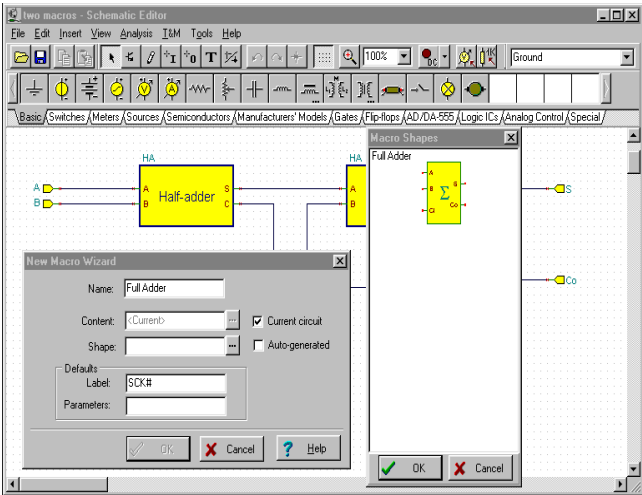


To do this, insert the newly created Half Adder twice into a new circuit and then add the additional components and wires as shown in the following picture.



Now create and save the new macro with the *New Macro Wizard* from the *Tools* menu. At this point let us note that although the automatic symbol creation is very convenient, you can also create your own schematic symbols with *TINA4*'s Schematic Symbol Editor and assign macros to them. Let us use this feature with an existing symbol. The creation of such a symbol will be described later in detail.

Set the Name to Full Adder and set the Label to FA (this will be displayed as the component label above the shape). Uncheck the Auto generated checkbox and press the  button next to it. The list of available symbols will appear as shown below.

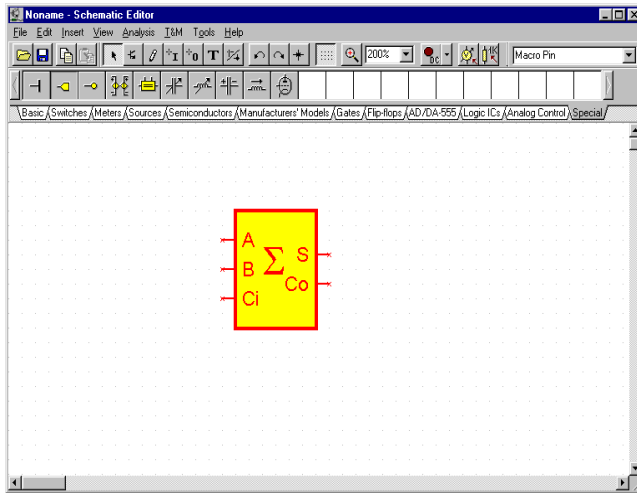


Note that in order to see the predefined symbols, the Macro Pin Label names must exactly match the names in the symbol. In our example, they must be (A, B, Ci, Co, S). If you do not see the symbol shown in the figure above, check the terminal names or try to recreate the symbol as shown later at “Making your own schematic symbols”.

Click the schematic symbol with the large summation sign and press OK. The name of the schematic symbol will appear in the shape field of the New Macro Wizard dialog box. Finally, click OK and save the macro under the name of Full adder.tsm.



To see our newly created full adder macro, select Macro from the Insert menu and then select Full adder.tsm and click Open.





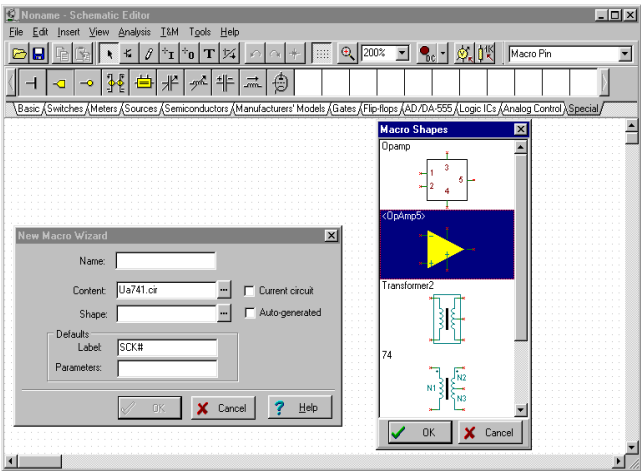
Double click on the symbol and the schematic inside the circuit will appear, showing two of our previous half adder macros. You can then double click on either of the macros to see the schematic inside. Return to the Full Adder and the main circuit by clicking the Leave Macro button.

## 5.2 Making a Macro from a Spice subcircuit

In *TINA*, you can create your own components from any Spice subcircuit that you have made or downloaded from the Internet. Note that there are also Spice component models in the large and extensible manufacturers' model library provided with *TINA*. The extension of those libraries is described later.

Let's create a UA741 operational amplifier using a Spice subcircuit. First, select the file containing the subcircuit.

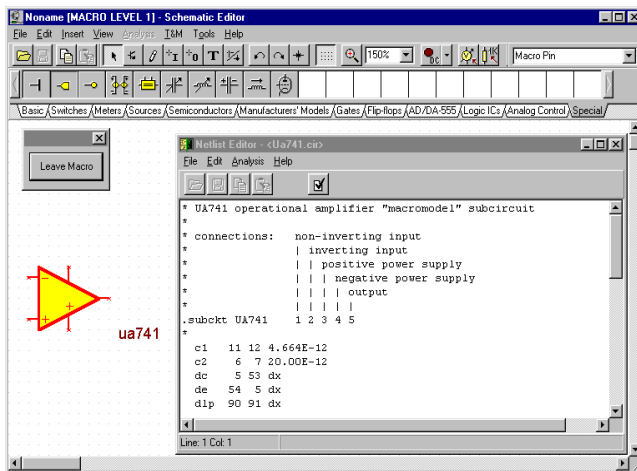
To do this, select the New Macro Wizard from the Tool menu. Uncheck the Current Circuit checkbox and press the  button. An Open dialog box will appear. Since we want to read in a Spice file, set the "Files of type:" field at the bottom of the dialog box to "Spice circuit file." Select the Examples\Spice folder, then select the UA741.CIR file, and press the Open button. We could use the automatic shape generation option (in which case a rectangular shape would be automatically generated), but let's select a more appropriate operational amplifier symbol instead. Uncheck the "Auto generated" checkbox and press the  button. The list of available symbols will appear.



Select the triangular *opamp* symbol and press *OK*. A Save dialog box will appear. Save the new Spice macro under the name UA741.CIR.

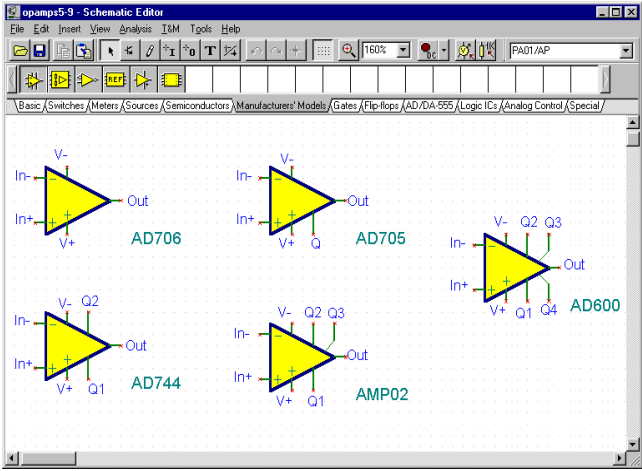
Now let's insert the new subcircuit into a schematic and check its contents.

Select the Macro command of the Insert menu. Click the UA741.tsm file and then press Open. Now the new macro will be attached to the cursor. Position it on the screen and drop it by clicking the left mouse button. Double click the symbol to see its content. The netlist editor will appear, showing the macro in detail. Note that you can modify this netlist, and the modified netlist will be saved with your circuit. However this will have no affect on the original macro; it remains unchanged.



#### NOTE:

It is very important that the connection names in the Spice macro match the pin names of the component symbols. The pin names of the opamp symbols predefined in *TINA* are shown below. You can check the pin names of a component symbol in *TINA* by moving the cursor above the pin. The pin name will be displayed at the left side of the bottom status line. Double-clicking the symbol will present the netlist, where you can check the connection names in the Spice subcircuit.



## 5.3 Using and extending Manufacturers' Spice model catalogs in TINA

In *TINA* you will find large catalogs of manufacturers' Spice models. In most cases, you will find that the required components are already in *TINA*'s manufacturers' model catalog. You can select components by function, manufacturer and part number.

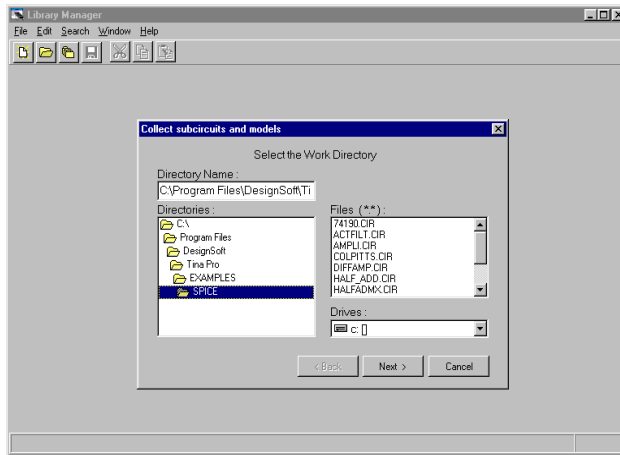
You can also extend the library using *TINA*'s Library Manager program.

### 5.3.1 Using the Library Manager

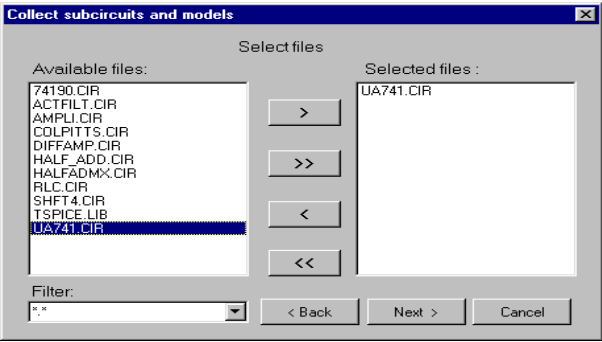
*TINA* has large libraries containing Spice models provided by semiconductor manufacturers such as Analog Devices, Texas Instruments, National Semiconductor, and others. You can add more models to these libraries or create your own Spice Library using *TINA*'s Library Manager (LM).

Let's learn how to add a Spice model to *TINA*'s Spice libraries:

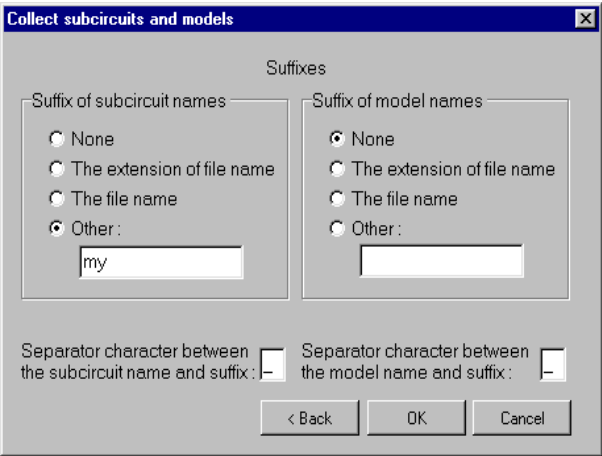
First, start the Library Manager program. Use the Windows 95 Start menu to locate the *TINA* folder and double click on its icon. Select *Collect subcircuits and models* from the *File* menu. Find the *EXAMPLES\SPICE* folder in the dialog box, click the *SPICE* folder where our example subcircuit -a ua741 amplifier- has already been placed, and press Next.



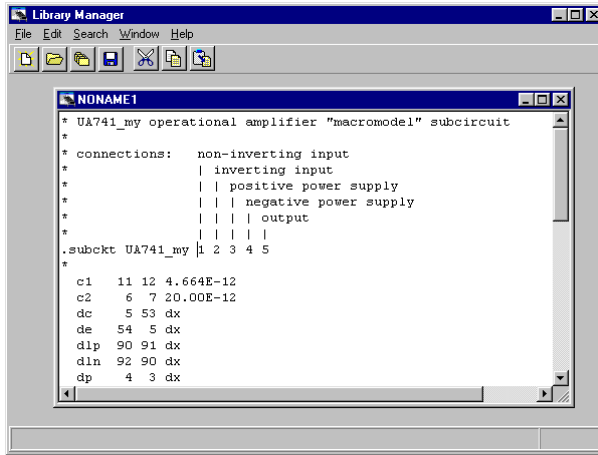
A new dialog box will appear with the list of available files on the left side. Note that the file you select must be a proper Spice subcircuit. Click UA741 and then press the > button. The UA741 model you have selected will appear on the list of selected files. In a similar way, you can select more files or even all the files by pressing the >> button.



Press the Next button to continue. The following dialog box will appear.



With this dialog box you can make changes in the subcircuit or model name. This might be necessary to avoid conflicts among different subcircuits or model versions with the same name. To differentiate the new model, you can add the file name as a suffix to the subcircuit name or add any text as a suffix to the name using the Other option. Let's add the suffix "my" to the subcircuit name and then press the Next button. The contents of the new library file will appear.

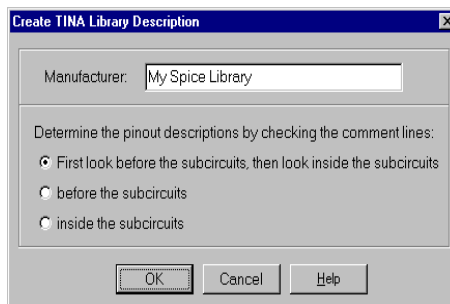


**NOTE:**

The new name of the subcircuit has the suffix "my": UA741\_my.

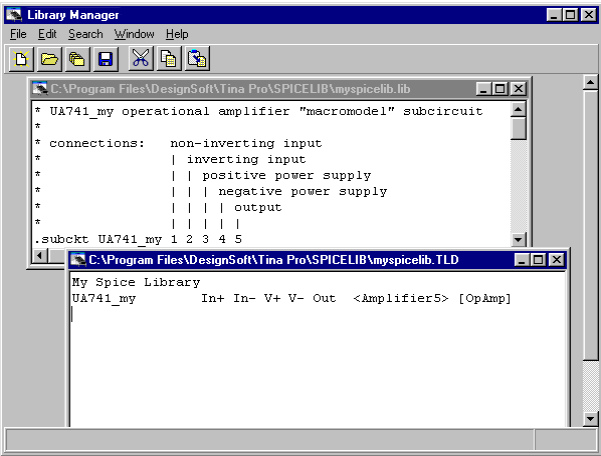
Using *File | Save As*, save this library in the SPICELIB folder to be found in the main TINA folder (e.g., **TINA 7\SPICELIB**), with the name *myspicelib.lib*.

Now select *Create TINA Library Description* from the *File* menu. The following dialog box will appear.

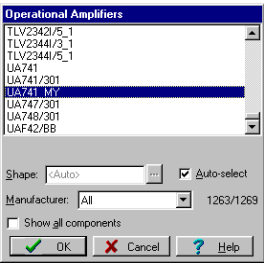


Here you specify a name for your new library. You can also specify some search options for determining the pinout description of the Spice model. The default setting is usually satisfactory. Press the Help button for more information.

The description (directory) of the new catalog will be displayed in a new window.



Finally, save the library directory as `myspicelib.tld` in *TINA*’s Spicelib folder. Note that the Save As command applies to the active (selected) windows only..

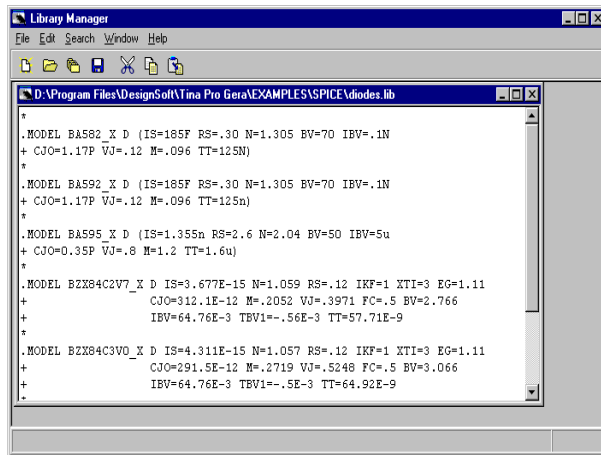




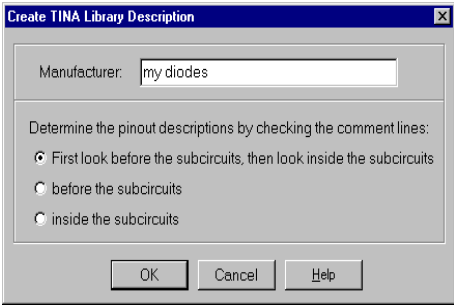
Next time you start *TINA*, select Spice Macros, and then Operational Amplifiers, and you will find the new component library in the list of Manufacturers. Your subcircuit will appear on the list invoked either by selecting “My Spice Library” or All.

In the previous example, you added a component described by a Spice subcircuit. You can also add diodes, transistors and other devices by simply using .MODEL instructions. These devices are normally placed in a file containing many .MODEL instructions. In *TINA*, there are two such sample libraries, called diodes.lib and transistors.lib

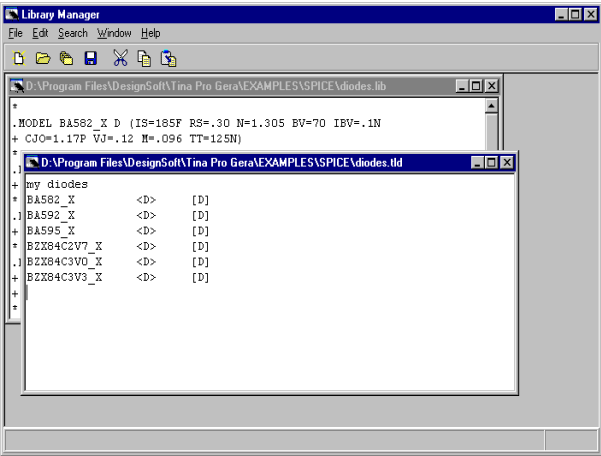
First, open the diodes.lib file from the EXAMPLES\SPICE folder using the File/Open File command or the corresponding icon on the toolbar. The following window listing the contents of the file appears:



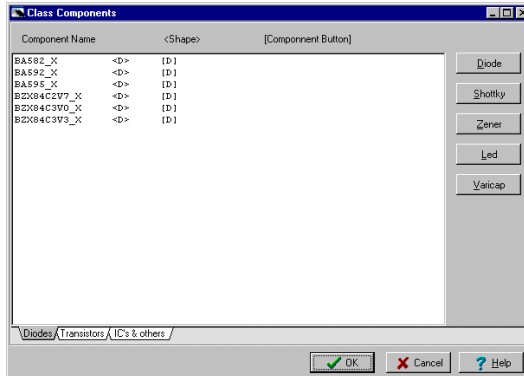
Do not change the other settings. Press OK. A description listing models in the new *TINA* library will appear:




Do not change the other settings. Press OK. A description listing models in the new TINA library will appear:

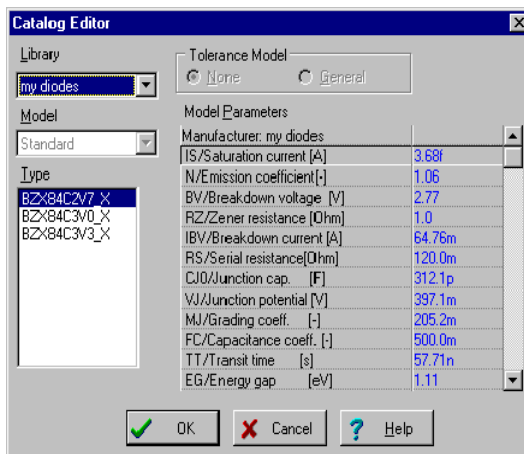


The file contains 3 normal and 3 Zener diodes. In the Spice language, there is no difference between normal, Zener, LED, Schottky, Varicap, and other diodes. However, in TINA, you can assign different schematic symbols to these types. To do this, select Categorize Components from the Edit menu. The following dialog box will appear:



Select the Zener diodes (the last 3 items on the list) by clicking them one by one while holding the Ctrl key. Then press the Zener button. <D> and [D] identifiers will change to <DZ> and [DZ] ensuring the use of the Zener diode symbols in TINA. Press OK and save both library files in the SPICELIB folder in the main TINA folder.

To verify the new diodes, restart TINA, select diodes or Zener diodes from the toolbar, drop the diode onto the schematic, and double click on it. Press the  button at the type line and select the "my diodes" library using the drop down menu of the Library field at the left top corner of the Catalog Editor dialog box.



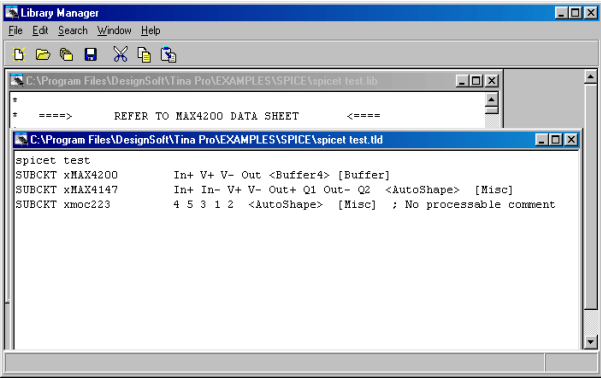
You will also find your new "normal" diodes under the Diodes category of the toolbar. Note that you can add new diodes to any existing manufacturer catalog if you select a library name already in the drop down menu (e.g., Motorola).

In a similar way, you can try adding transistors given by .MODEL Spice commands to TINA using the transist.lib library. There is no need to categorize these components since the NPN and PNP transistors have different notations in Spice.

In many cases, adding models to TINA is as easy as described above, but in some cases TINA cannot make the connection between the Spice models and their graphic symbols automatically. Fortunately, TINA's latest Library Manager provides an easy to use tool to solve this problem.

Let's the add the library "SPICETST.LIB" from the EXAMPLES\SPICE folder to TINA.

First, start the Library Manager as described above. Open the "SPICE TEST.LIB" file using the Open icon or the Open command from the File menu. Use the "Create TINA Library Description/for Spice Models and Subcircuits" Command as described above. You will see the following window.



Looking at the lines of the "spice test.tld" file containing the library description, it seems that the first model, xMAX4200, was recognized automatically, since both the graphic symbol and the category were found.

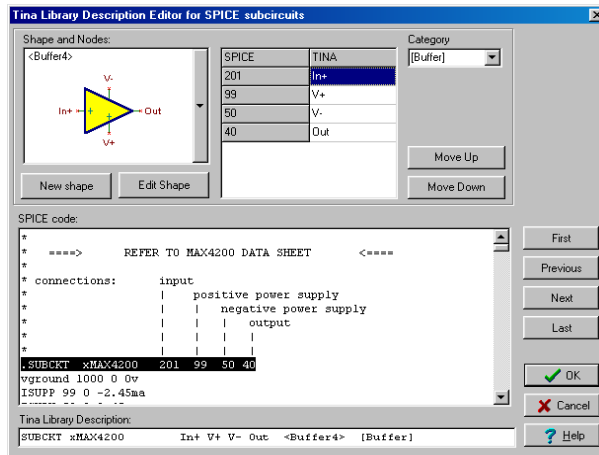
For the second model, the xMAX4147, no graphic symbol was assigned and its category was not recognized; however, the program recognized its terminals.

Finally for the last model, the xmoc223, nothing was recognized.

Note that even if the program does not recognize a model, it is placed in an automatically generated box (autoshape) and still can be used.

If you want to add an appropriate graphic symbol to such a part, you should do the following:

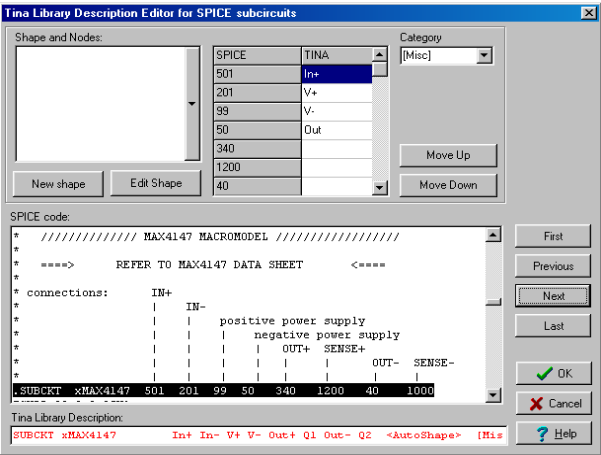
Select the "Edit/TLD Editor for Subcircuits" command. The following dialog box will appear:



TINA presents the graphic symbol with the names of the terminals in the top left corner. In the top right corner, you can see the list of terminal nodes and the associated terminal name of the graphic symbol. You can move the graphic terminal names up or down by simply dragging them or by using the Move Up and Move Down buttons. In the top right corner, TINA displays the category, which you can also change.

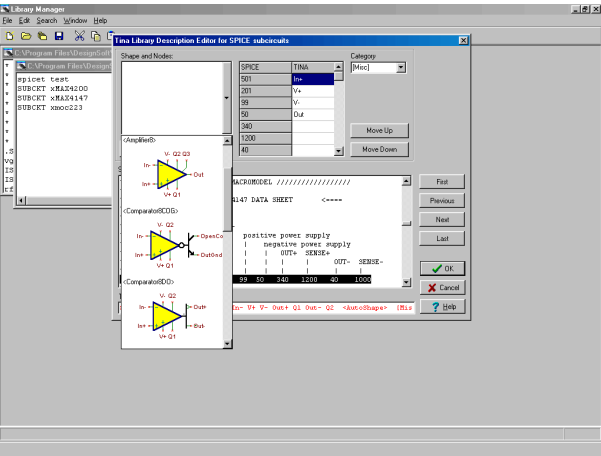
Next, the SPICE code of the selected component is shown, while below that in the "TINA Library Description" field appears the content of the actual line of the .TLD. With the buttons First/Previous/Next/Last you can move among the models in the library.

Verify that all entries for the first model are correct and jump to the second model by pressing the Next button. In the dialog box, the MAX4147 model will appear.

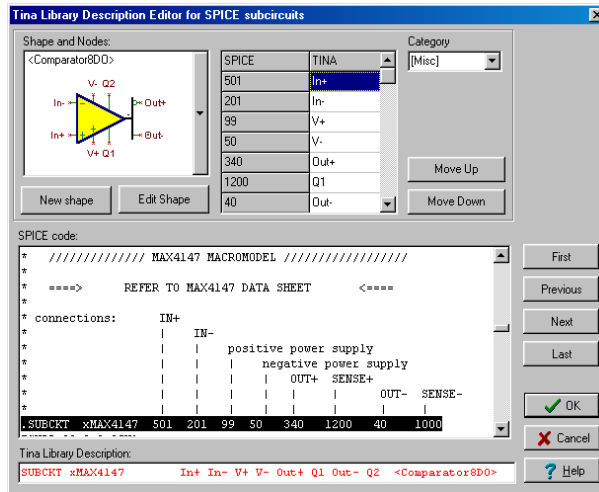


The first things you should notice are the missing graphic symbol (Shape) at the top-right corner and the red TLD line at the bottom. These indicate that the Library Manager could not fully understand the model.

Press the vertical button at the right side of the Shape window and see a list of the available symbols appear. Move down with the scroll bar until you see the <Comparator8DO> "totem pole" symbol.



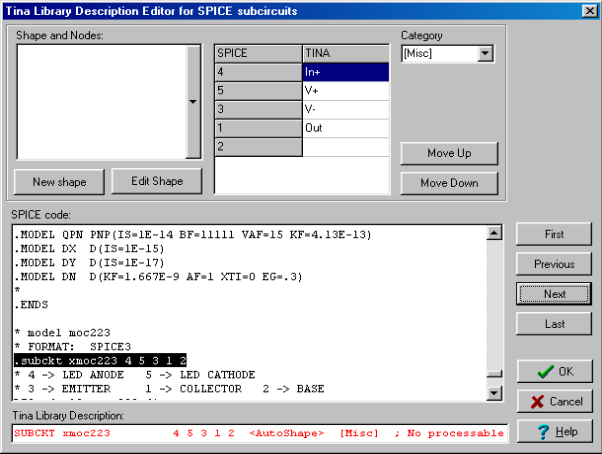
Click the <Comparator8DO> "totem pole" symbol. This symbol will be inserted into the Shape window.



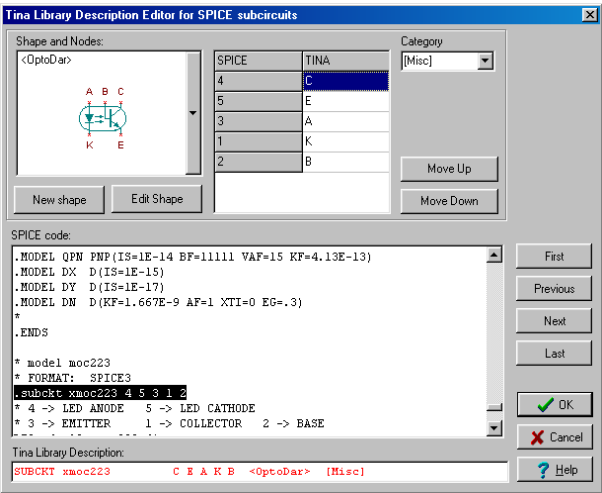
Check the list showing the connection between the Shape nodes and the Spice terminals. It should be correct since the Library Manager did not give an error message for the Spice comments. If there had been a discrepancy, you would have seen an error message in the TLD line: "No processable comments".

Change the Category at the top right corner to <Comparator >.

Click the Next button again to bring in the last model in this library. The following window will appear:



Note the "No processable comments" message at the bottom line. This means that the Library Manager could not identify the Spice terminals from the comments in the Spice model. We will have to make the connection between the graphic shape and the Spice terminals manually.





First select the <Optodar> shape as described above. You will see the following window:

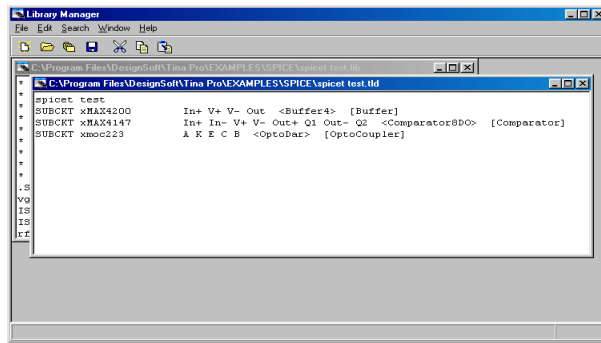
Note that the Spice terminal nodes and the Shape terminals are not properly mapped. The first node in the list, node 4, is associated with the Collector of the transistor while according to the comments in the Spice code it should be connected with A, the Anode of the input LED.

Click on A and drag A to the top of the list (to Node 4) , then drag K to Node 5, and finally drag E to node 3.

Check the remaining two nodes, which in this case must be correct.

Change the Category at the top right corner to <Optocoupler >.

This completes are editing of the various models. Press OK to close the TLD editor. The SPICE TEST.TLD window should be updated and look like this:



Using the File|Save As command, save both the SPICE TEST.TLD and the SPICE TEST.LIB files in the TINA 7\SPICELIB folder. It is important to use this folder and not the EXAMPLES/SPICE folder, otherwise TINA will not see the changes.

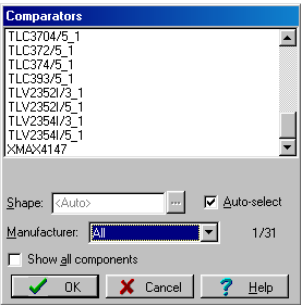
Finally, use the File|Create TINA Library command to register the changes for TINA.

Close the Library Manager.

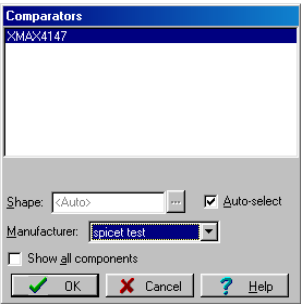
When you restart TINA, you can find the new models by selecting "Spice Test" as Manufacturer at any category of the Spice Macro Models component bar and by setting the "Show all components" checkbox.



You can also find these new models by looking in the appropriate category (Comparators, Buffers, and Optocouplers). These new parts will be at the end of the list, since the names of the new models start with X.



Of course, you can also set the Manufacturer in the appropriate category to "Spice test" to see the newly added components.



## 5.4 Adding S-parameter models

Let's learn how to add an S-parameter model to TINA's libraries.

Start the Library Manager program. Use the Windows Start menu to locate the TINA PRO folder and double click on the Library Manager icon.

Next, select Collect S parameter files... from the File menu. *Note: You should not use the File|Open... command to collect S parameter files.*

Find the folder EXAMPLES\RF in the dialog box and double-click on the SPICE folder. This is where our example, an S-parameter transistor called s\_bfp405.s2p, has already been placed. Press the Next button. Note: the S-parameter files must have either S1P or S2P extension (the manufacturers use the same convention). If the extension is S1P it means that the device is a "1-port" (described with 1 parameter) otherwise a "2-port" device (described with 4 S-parameters).

A new dialog box will appear with the list of available files on the left. Note that the file you select must be a proper S-parameter file. S-Parameter data files are in the TouchStone format. This is a typical data segment of a two-port file:

### S-Parameter file description

```
# MHz S R I R 50
0.30 0.02 -0.05 -0.03 -0.02 -0.03 -0.02 0.02 -0.05
0.31 0.03 -0.06 -0.02 -0.01 -0.02 -0.01 0.03 -0.06
0.33 0.04 -0.07 -0.01 -0.03 -0.01 -0.03 0.04 -0.07
....
```

The first line is a header that gives the frequency units, parameter, measurement format, and characteristic impedance of the measurement (here, 50 Ohms).

The first column is the frequency in Hz. The next columns are, in order, S11 Real, S11 Imaginary, S21 Real, S21 Imaginary, S12 Real, S12 Imaginary, S22 Real, S22 Imaginary. One-port data files are similar to the two-port files, except that there are no columns for the S21, S12 and S22 parameters.

Click on `s_bfp405.s2p` and then press the `>` button. The `s_bfp405.s2p` model you have selected will appear in the list of selected files. In a similar way, you can select more files or even all the files by pressing the `>>` button. A dialog appears. With this dialog box you can change the model name. This might be necessary to avoid conflicts among different model versions with the same name. To differentiate the new model, you can create a model name from the file name or from one of the first 8 lines, or you can add a prefix or suffix to the model names. Let's just use the file name as a model name. Press the Next button and the contents of the new library file will appear.

Using File|Save As, save this library in the SPICELIB folder found in the main TINA folder (e.g., TINA PRO\SPICELIB), using the name `myslib.lib`. Now select Create TINA Library Description...|...for S parameter models from the File menu. The following dialog box will appear:

Here you specify a name for your new library, e.g., My S Parameter Library. You could specify the name of the manufacturer as a library name, but note that if there already is a library in TINA with the same name (e.g., Siemens), then your new model will be added to this library. The library descriptor file of the new catalog will be displayed in a new window. However, in the case of S parameter files, you must always categorize the models (unless you want them to appear among the mixed components with a default shape). To do this, select Categorize Components from the Edit menu. The following dialog box will appear:

Press the IC's & other unrecognized components tab. Select one or more models from the list, then press a Move to page ... button (pick the button for the model type of the selected model). In our case, press Move to page Transistors, then click on the Transistors tab. Now select the appropriate category, which for this model is NPN.

Save the library descriptor file as `mysplib.tld` in TINA's SPICELIB folder. (Both Spice and S parameter libraries are stored in this folder.) Note that the Save As command applies to the active (selected) windows only.

Finally, use the File | Create TINA Library command to register the changes for TINA.

Next time you start TINA, select RF components, and then NPN RF Bipolar Transistors and you will find the new component library in the list of Manufacturers. Your S parameter model will appear on the list invoked either by selecting “My S Parameter Library” or All.

## 5.5 Making a VHDL macro from a .vhd file

You can create a VHDL macro from any .vhd file that contains an entity (interface to the outside world) with its architecture (description of the hardware). The ports declared in the entity will automatically appear in the macro. By default, the input ports of the entity will appear on the left side of the generated macro shape and the output ports of the entity will appear on the right side, but by editing the generated macro you can change this arrangement.

For example:

```
ENTITY e_Half_add_entity IS PORT(
    A : IN std_logic;
    S : OUT std_logic;
    C : OUT std_logic;
    B : IN std_logic );
END e_Half_add_entity;
```

In this case the A,B ports will appear on the left side and the S,C ports will appear on the right side.

Lets see how to do a macro from the following VHDL code (a half adder):

```
LIBRARY ieee, tina;
use ieee.std_logic_1164.all;
use std.textio.all;
USE tina.primitives.all;

-----
- entity section
-----

ENTITY e_Half_add_entity IS PORT(
```

```

A : IN std_logic;
S : OUT std_logic;
C : OUT std_logic;
B : IN std_logic );
END e_Half_add_entity;

-----

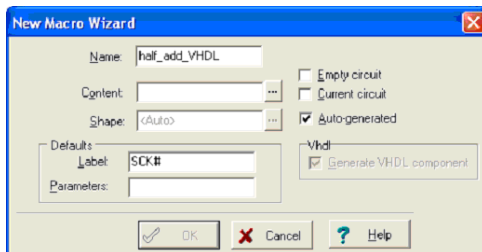
-- architecture section


-----

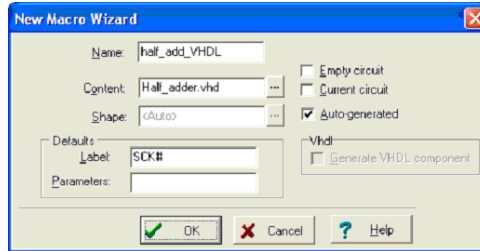
ARCHITECTURE a_Half_add_arch of e_Half_add_entity IS
constant delay : time := 20 ns;
BEGIN
    S <= (A xor B) after delay;
    C <= (A and B) after delay;
END a_Half_add_arch;

```

1. Select Tools/New Macro Wizard...
2. Check “Generate VHDL Component”, uncheck “Current Circuit.”
3. Type a name for the new macro.



4. Click on the  button in the content line, select the VHDL file type, and navigate to EXAMPLES/VHDL in the TINA program folder. You should see the half\_adder.vhd file in the open dialog. Select this file and press Open.
5. Now the “New Macro Wizard” dialog should appear looking like this:



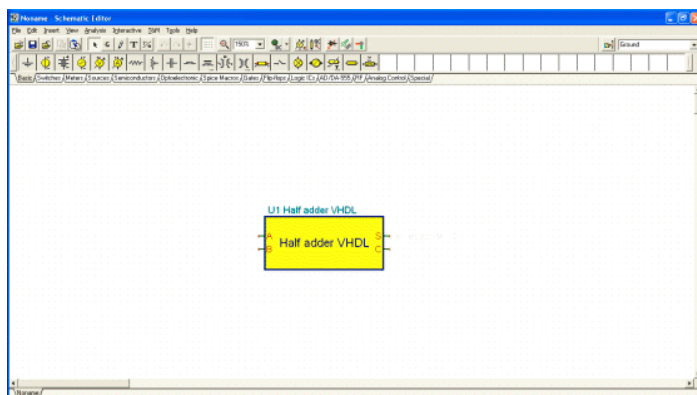
- Press OK to save the macro, and save the macro into the default Macrolib folder.

### 5.5.1 Placing a VHDL macro in the schematic editor

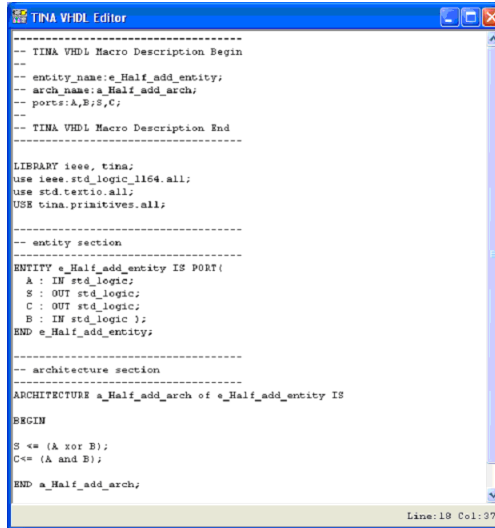
Not let's see how can we insert our previously saved VHDL macro into TINA's schematic editor.

- Select "Insert/Macro..." from the menu and select the previously saved macro half\_add\_VHDL.TSC from the MACROLIB folder of TINA's main program folder.

The screen will look like this:



To see the content of the macro double-click on it and press the Enter Macro button on the property dialog that appears. The VHDL content of the macro will be displayed:



```

-----
-- TINA VHDL Macro Description Begin
--
-- entity_name:e_Half_add_entity;
-- arch_name:a_Half_add_arch;
-- ports:A,B;S,C;
--
-- TINA VHDL Macro Description End
-----

LIBRARY ieee, tina;
use ieee.std_logic_1164.all;
use std.textio.all;
USE tina.primitives.all;

-----
-- entity section
-----
ENTITY e_Half_add_entity IS PORT(
  A : IN std_logic;
  S : OUT std_logic;
  C : OUT std_logic;
  B : IN std_logic );
END e_Half_add_entity;





-----
-- architecture section
-----
ARCHITECTURE a_Half_add_arch of e_Half_add_entity IS

BEGIN

  S <= (A xor B);
  C<= (A and B);

END a_Half_add_arch;
  
```

## 5.5.2 Testing a VHDL macro

Let's test our newly created macro in TINA's VHDL interactive mode. To do this, place two High-Low digital switches from the Switches toolbar, one for each of the A,B inputs, and two logic indicators from the Meters toolbar. Now select the VHDL interactive mode with the  button or from the Interactive menu and press the  button. The logic levels of the nodes will appear, Red for High. Blue for Low. The logic indicators will also show the logic level of the outputs in a Red square  for High, and empty square  for Low.



### 5.5.3 Changing the pin arrangement of a VHDL macro

To change the pin arrangement, you should add a special header to your VHDL macro.

The easiest way to do this is to open the automatically generated macro and edit its header.

For example the header in the previous example is

```
-----
- TINA VHDL Macro Description Begin
- entity_name:e_Half_add_entity;
- arch_name:a_Half_add_arch;
- ports:A,B;S,C;
- TINA VHDL Macro Description End
-----
```

The pin arrangement is determined by the:

```
ports:A,B;S,C;
```

line, the ports before the first semicolon (;) are placed on the left while the rest are placed on the right side of the macro box.

For example, if you change the ports line to

```
ports:A,B,S;C;
```

and add the whole changed header to the original VHDL file (which had no header) we get the following file (you can also load it from the EXAMPLES/VHDL/half\_adder31.vhd)

```
-----
- TINA VHDL Macro Description Begin
- entity_name:e_Half_add_entity;
- arch_name:a_Half_add_arch;
- ports:A,B,S;C;
- TINA VHDL Macro Description End
-----

LIBRARY ieee, tina;
use ieee.std_logic_1164.all;
use std.textio.all;
```

```

USE tina.primitives.all;

-----

- entity section
-----

ENTITY e_Half_add_entity IS PORT(
    A : IN std_logic;
    S : OUT std_logic;
    C : OUT std_logic;
    B : IN std_logic );
END e_Half_add_entity;

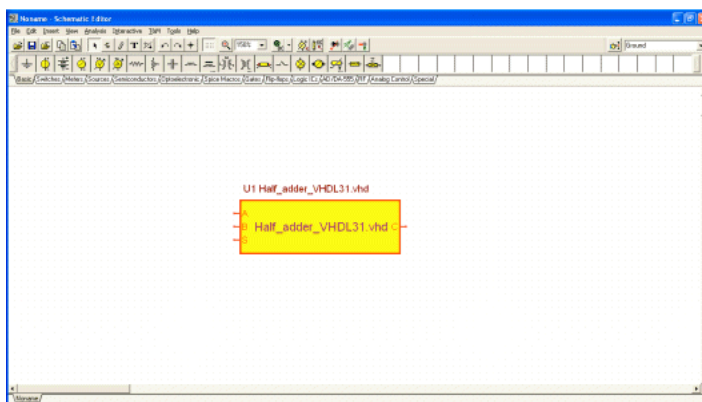
-----

- architecture section
-----

ARCHITECTURE a_Half_add_arch of e_Half_add_entity IS
BEGIN
    S <= (A xor B);
    C<= (A and B);
END a_Half_add_arch;

```

Converting this into a new macro called Half\_adder\_VHDL31.TSM and then inserting it again we will see the revised pinout version::



# MAKING YOUR OWN SCHEMATIC SYMBOLS AND FOOTPRINTS

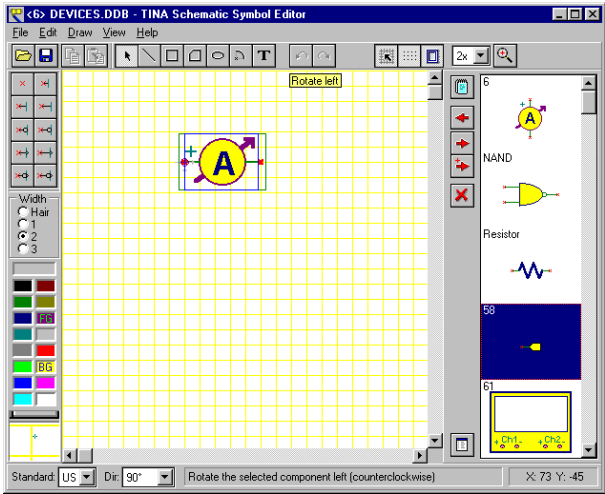
## 6.1 Schematic Symbol Editor


---

Using *TINA*'s Schematic Symbol Editor, you can create new schematic symbols so that you can add your own circuit components to *TINA*.

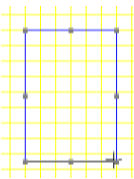
To create new symbols, you place lines, arcs, rectangles, and arbitrary characters with any fonts, specifying line-width, color, and area color fills. After drawing the symbol, you add and define connections to it.



To get acquainted with some features of the editor, read in the list of existing symbols. Use the Windows XP (also versions 98SE and 2000) Start menu to locate the *TINA PRO* folder. Start the Schematic Symbol Editor by clicking its icon, then Select *File | Open* and double click on the *devices.ddb* file. On the right hand side of the Editor Window, the list of current schematic symbols will appear.

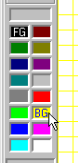


The first symbol on the list (an Ammeter) will appear in the editor window. Try the **Dir:** control at the bottom of the screen. Using this control, you can provide different shapes for symbols at each rotational orientation by designing each of them individually. Now click on the NAND symbol at the right side of the screen and press the  button. The NAND gate symbol will appear in the editor window. Try the **Standard:** control to see the US and European versions of the part shape. You can design symbol versions for each standard, if necessary. If the symbols are identical in the two standards, you need create only one version.

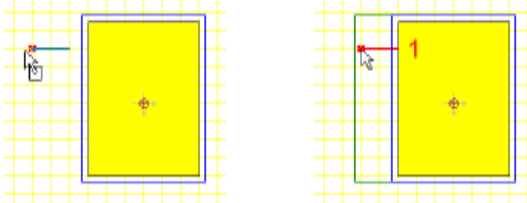
Now let's create a symbol for the full-adder circuit that was used above in our example of creating a half adder macro.



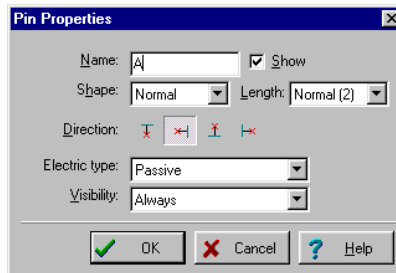
First clear the editor window by pressing the  button or by selecting the actual symbol in the editor window and pressing the Del button. Now draw a rectangle as the body of the component. First press the  button then click on any point in the drawing area, hold the mouse button, and move the mouse until the rectangle is properly sized.



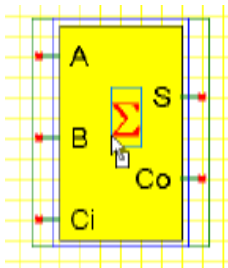
Fill the rectangle with a color by right-clicking on the palette at the lower left corner of the window. Note that a left-click will change the foreground color (FG), in our case the border of the rectangle.



Now add the terminals. Select the desired terminal type from the Terminal-Toolbar in the upper left corner of the window and move the cursor into the rectangle you just drew. Position it using the mouse or pressing the  $[+]$  or  $[-]$  key for rotation and click to locate the terminal. Be sure the small red x, indicating the pin end, is outside the body. Continue this process until every terminal is positioned.



## Wizard in the Schematic Symbol Editor

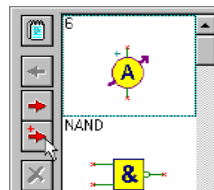


After you have positioned all the terminals, you can establish their properties by double clicking on each of them.

You should assign terminal names as shown in the picture below.

Next, add a large summation sign. Click on the **T**-Text-Editor button on the Toolbar,

enter an S in the window, and select a font. To get the special Greek summation sign, select the Symbol-Font.

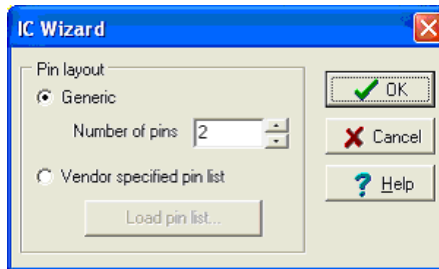


Press the **D**-Device Properties button, set the Name of the symbol to Full Adder, and press OK.

Finally, copy the new symbol into the symbol library with the **R** button (it appears now at the end of the list), and use the File|Save command to save the now extended devices.ddb file in the *TINA* main directory.

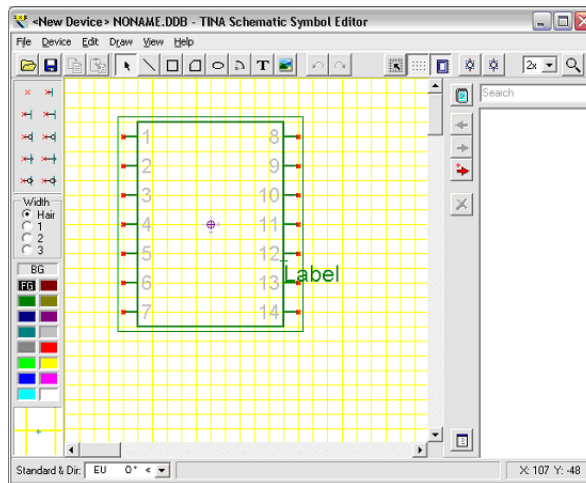
## 6.2 IC Wizard in the Schematic Symbol Editor

When you need to create the shape of an IC with a lot of pins, the IC Wizard can assist you. The IC Wizard can be activated from the Draw menu by selecting the IC Wizard command. The following dialog box appears:



The wizard offers two options.

- **Generic** If you select this option, the Wizard creates a rectangular-shaped IC with a DIP-style pin layout. The total number of pins must be specified. For example, if you enter 14 pins in this field, you get the following pin layout:

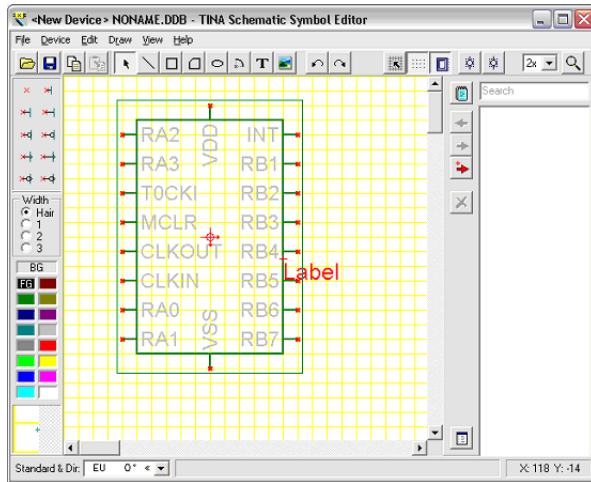


- **Vendor specified pin list** In this case the Wizard creates a shape based on a file where each line defines a terminal as **Pin number, Name, Electric type** separated by commas:

For example..

- 1,RA2,INPUT
- 2,RA3,INPUT
- 3,RA4/T0CKI,INPUT
- 4,MCLR,INPUT
- 5,VSS,POWER
- etc.

The electrical type can be INPUT, OUTPUT, INOUT, BUFFER and POWER. For example, if you read in the PIC16F84A.CSV file from TINA's EXAMPLES\PCB folder, the Wizard generates the next IC:



When the Wizard is finished, the shape can be further edited with the tools described above.

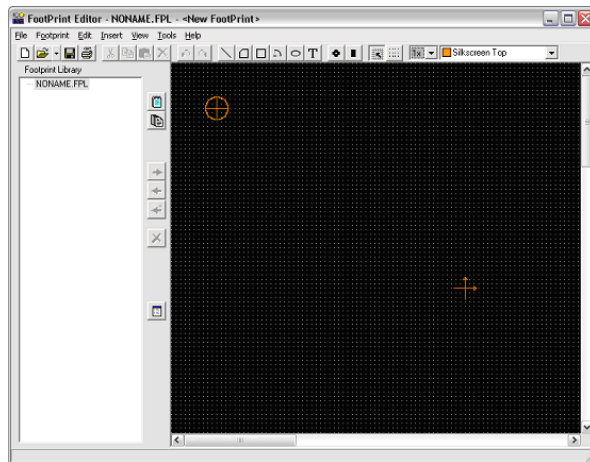
## 6.3 Footprint Editor

Using the Footprint Editor, you can create new footprint symbols that you can add to the footprint library. You can start the footprint editor from the Tools menu of TINA's PCB Designer by selecting the Footprint Editor command.



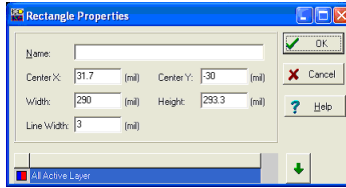
If you want to create a new footprint, you can build it by placing various primitive drawing elements and symbols, including lines, rectangles, arcs, text and pads. We'll recreate a simple resistor footprint already included in the system.

First clear the editor window by selecting the New Footprint command from the Footprint menu. Then set the position of the origin by double clicking on the cross symbol with the little arrows. Enter 1300, 1000 in the X and Y fields respectively. Check the Use Relative Coordinates checkbox, and press OK.



Now select the rectangle symbol on the toolbar and draw a rectangle around the origin. To do this, click on one corner, hold down the mouse left button, and drag the cursor to the opposite corner. Release the mouse button. If you create a footprint, you should be very careful with the dimensions. You must define the exact dimensions according to the manufacturer's data sheet, especially of the pads: otherwise the parts cannot fit on the board. To set the shape precisely, it is better to use coordinates rather than drawing with the mouse.

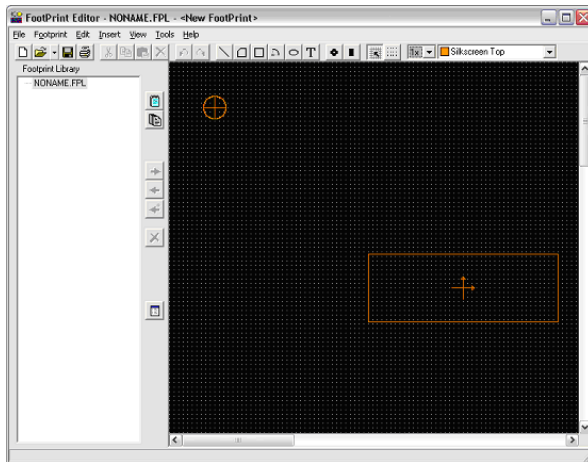
To set the size of our rectangle using coordinates, move the mouse over one of its edges and when the cursor changes into a hand symbol, double click at one edge of the rectangle. The Rectangle Property dialog will appear.



Now enter 0, 0 in the CenterX and CenterY fields; 840, 300 into Width and Height; and 5 into the Line Width fields..

In the Rectangle Property dialog of the shape you can change the layer settings, too. By default, a rectangle shape resides on the Silkscreen Top and Assembly Drawing Top layers.

Pressing the down arrow invokes the layer configuration editor. The layers can be turned on/off by double-clicking on the grey square next to the layer name. In our example, the default layer configuration is good, so don't change it. Close the property editor by pressing OK.



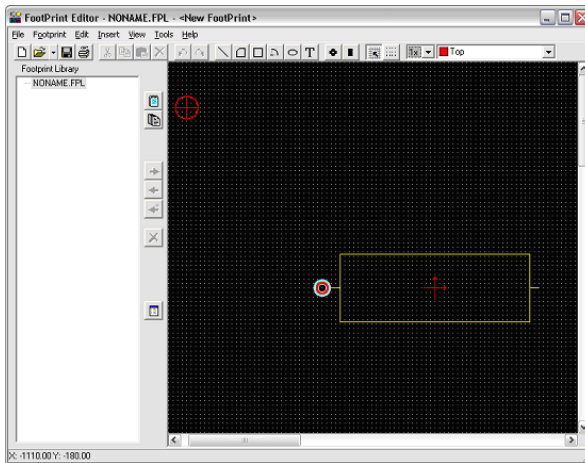
Now add 2 lines to our footprint. Select the line symbol and draw 2 horizontal lines next to the rectangle on both sides. Double click on the lines and modify the parameters as follows:

*Line1:* -460, 0, -420, 0, 5 (Point1 X, Point1 Y, Point2 X, Point2 Y and Line width)

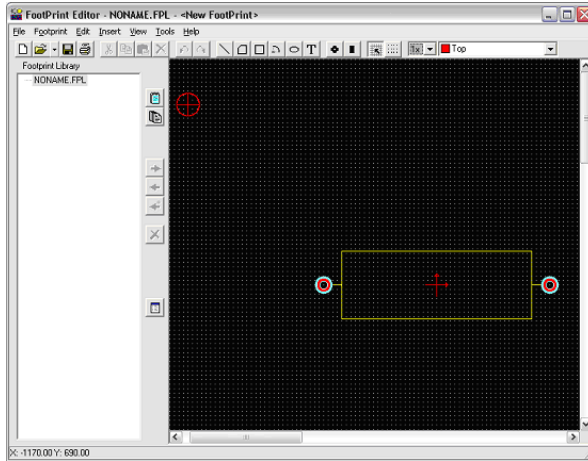
*Line2:* 420, 0, 460, 0, 5 (Point1 X, Point1 Y, Point2 X, Point2 Y and Line width)

Finally, add two through hole pads to the footprint symbol. Select the pad symbol from the toolbar. Move the pad next to Line1. Now activate the property editor of the pad, by moving the mouse over it and double-clicking when the mouse changes into a hand symbol. Enter - 500, 0 in the Center X and Center Y fields. The Drill parameter is 37. Now click on the down arrow. By default the pad resides on the Top, Bottom, Power, Ground, Solder Mask Top, Solder Mask Bottom, Drill Drawing and Drill Tape layers.

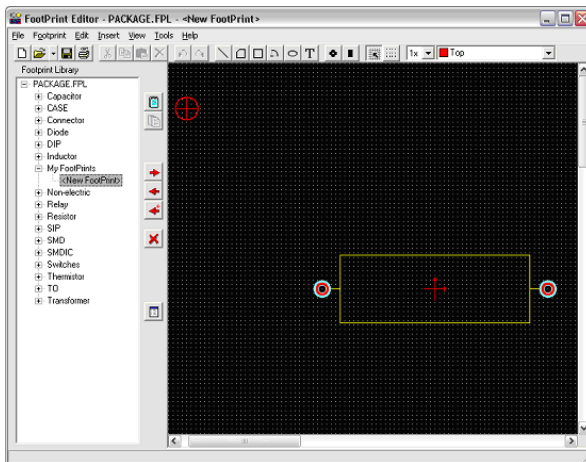
The default layer configuration could have been changed in a way similar to what we've seen in the rectangle example. Though the default layer configuration is good, we have to change the dimensions of the pad. Double click on the size field and enter 58 in the Diameter field on the Top, Bottom, Solder Mask Top and Solder Mask Bottom layers, enter 78 on the Power and Ground layers and 37 on the Drill Drawing and Drill Tape layers. It's important to enter the package pin number into the name field.



Now pick up the next pad and move to Line2. We have only one parameter to change, Center X, which should be 500.



The footprint symbol is ready to save into a library. Open the package.fpl file, select the resistor group (or define a new group) and press the add footprint button.



### 6.4 IC Wizard in the Footprint Editor

If you want to create the footprint of a more complex IC, e.g., an IC with a complex pin configuration, the IC Wizard can assist you. The IC Wizard can be activated from the Insert menu.

The wizard presents several properties of the IC which you can set.

In the **Technology group**, you can set the mounting mode and the package type of the IC. The mounting mode can be through hole or surface mounted. Depending on the mounting mode the following packages are available: DIP (Dual in line package), PGA (Pin grid array package), SPGA (Staggered pin grid array package), SOP (Small outline package), LCC (Leaded chip carrier package), QFP (Quad flat package), BGA (Ball grid array package), SBGA (Staggered ball grid array package), SIP (Single in line package) and ZIP (Zigzag in line package) respectively.

In the **Package dimension** group, the dimensions (length, width, 3D height) of the package can be set. Depending on the selected package, the 4<sup>th</sup> parameter is either notch, one, corner cutoff, or ignored.

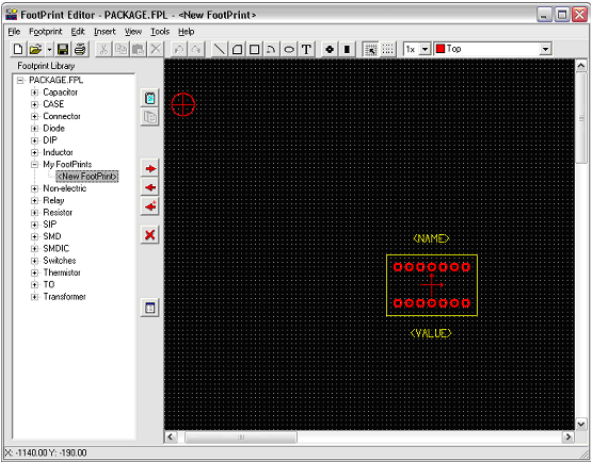
The **Pad dimension** defines the shape and dimensions (length, width) of the pad. If the mounting mode is through hole, the shape of the drilled pad can be round, square or octagon. Moreover, the shape and dimensions of the drill diameter can be defined. However, if the mounting mode is surface mounted, the shape of the pad can be circular, rectangular or rounded corner and the appropriate dimensions can be also set.

In the **Pad position**, the number of pins and the distances between them can be set according to the package type.

Finally, in the **Pad numbering** group, the type and direction for pad numbering can be entered, depending on the package type.

An example:

Technology: Through hole  
 Package type: DIP  
 Package dimension/Length: 400  
 Package dimension/Width: 270  
 Pad dimension/Shape: Round  
 Pad dimension/Drill hole: 20  
 Pad dimension/Diameter 40  
 Pad position/Number of horz. pins: 14  
 Pad position/Between pins: 50  
 Pad position/Between rows: 160



When the wizard is finished, the footprint can be further edited and saved in the library.

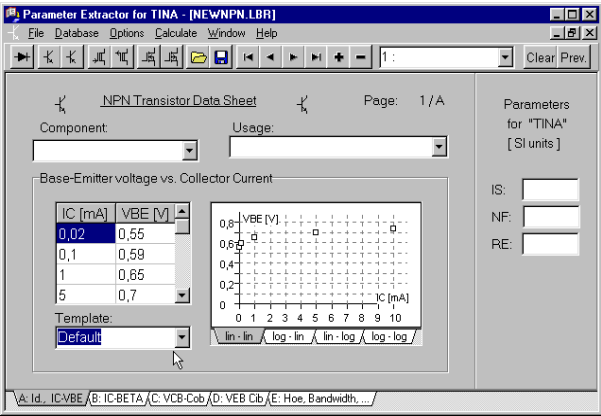
## CHAPTER 7

# USING THE PARAMETER EXTRACTOR

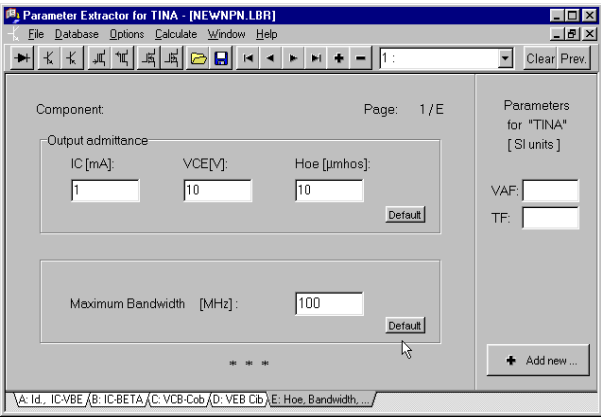
Using *TINA*'s Parameter Extractor you can create component models that more closely represent actual real world devices by converting measurement or catalog data into model parameters.



Use the Windows Start menu to locate the *TINA PRO* folder. Start the Parameter Extractor by double-clicking its icon. To create a new transistor, which can be added to *TINA*'s transistor catalog later, select *File* | *New Library* | *NPN Transistor*.

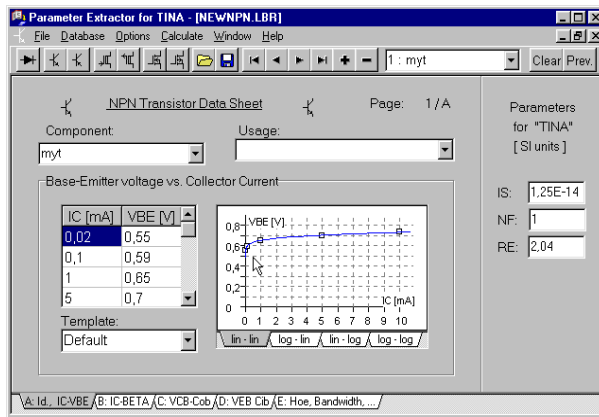


The dialog allows you to enter data from measurements, from manufacturers' catalog data, or from *TINA*'s default values (use the Template-ComboBox for this).



Click on each tab at the bottom of the screen and fill in all the transistor parameters. Select the default values or enter your own. Note: be sure to fill in all data, since missing data may lead to incorrect results.

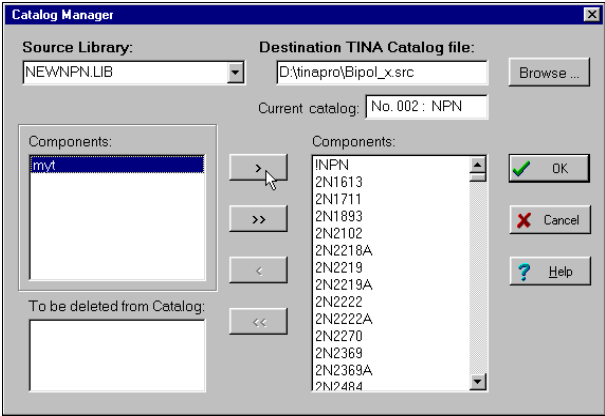





Next select *Calculate | Current component*. To check how well *TINA*'s transistor model matches the input data, you can walk through the tabs to see the calculated graphs and numeric values for every parameter.

Finally, let's insert the new transistor into the source file for *TINA*'s transistor catalog by selecting *File | Catalog Manager*. To be able to use the new catalog, you must recompile the modified source files and link them together into the CLCC.CAT catalog file.

Locate and open a component-file compatible with your component (e.g., if adding a bipolar transistor, choose a bipolar catalog, *bipol\_x.crc*). Click on the Browse-Button and select the file from the File Open Dialog. All component-files delivered with *TINA* are placed in the CLCC subdirectory of the *TINA* directory (by default *C:\Program Files\Designsoft\TINA7*).



Move your component into the library by selecting it, clicking on the  button and then the OK button.

After Pressing OK, TINA will prompt you and ask if you want to recompile the catalog source files and create a new updated catalog. If you answer "Yes," TINA will create the new catalog and you can use it after restarting TINA. You can also recompile the catalog using the "Compile TINA Catalog" command in the File menu. This may be necessary if a previous attempt at compiling failed, e.g., due to insufficient hard disk space.

In a similar fashion you can calculate magnetic core parameters. You should enter the upper ( A ) and lower ( B ) curve of the hysteresis and the geometric parameters of the core. Run an example with the default parameters ( load Default from the Template listbox ) to see typical values.

# ADVANCED TOPICS

## 8.1 Introduction

---

In the previous chapters, we have introduced TINA's main features and how they can be accessed. Our demonstration, however, is far from complete, for TINA contains many other useful and advanced features for designing, testing and teaching electronics and electronic circuits. These include S-parameter models, network analysis, details of Fourier series and Fourier spectrum analysis, symbolic analysis, post-processing of analysis results, creation of phasor diagrams, Nyquist diagrams, the built-in interpreter, multiparameter optimization, creation of multilayer PCBs and other topics.

The detailed description of these topics is not included in the printed Quick Start manual; they are published only in electronic form. They can be found on the TINA install CD and on the web at [www.tina.com](http://www.tina.com) in the Documentation section ( [www.tina.com/support.htm](http://www.tina.com/support.htm) ).

## 8.2 Table of Contents

---

- Parameter Stepping
- DC Transfer Characteristic and Parameter Sweeping
- Phasor Diagram
- Nyquist Diagram
- Noise Analysis
- Network Analysis and S-parameters
- Symbolic Analysis
- Post-processing Analysis Results
- Optimization
- Fourier Series and Fourier Spectrum
- Interpreter
- TINA PCB Design Manual

We are continuously adding new topics and examples to this chapter, so please check back regularly to the Documentation section of our website **[www.tina.com](http://www.tina.com)** for the latest version.